

■ PROJECTS ■ THEORY ■ APPLICATIONS ■ CIRCUITS ■ TECHNOLOGY

NUTS & VOLTS

NUTS & VOLTS

www.nutsvolts.com
November 2013

EVERYTHING FOR ELECTRONICS

Build The **CHROMATICON** *The Best Sound To Light Device You Could Ever Build!*

◆ **It's Open Mic Night!** ***How Does Your Guitar Sound?***

Fine-tune your guitar to sound great through just about any sound system with aux inputs.

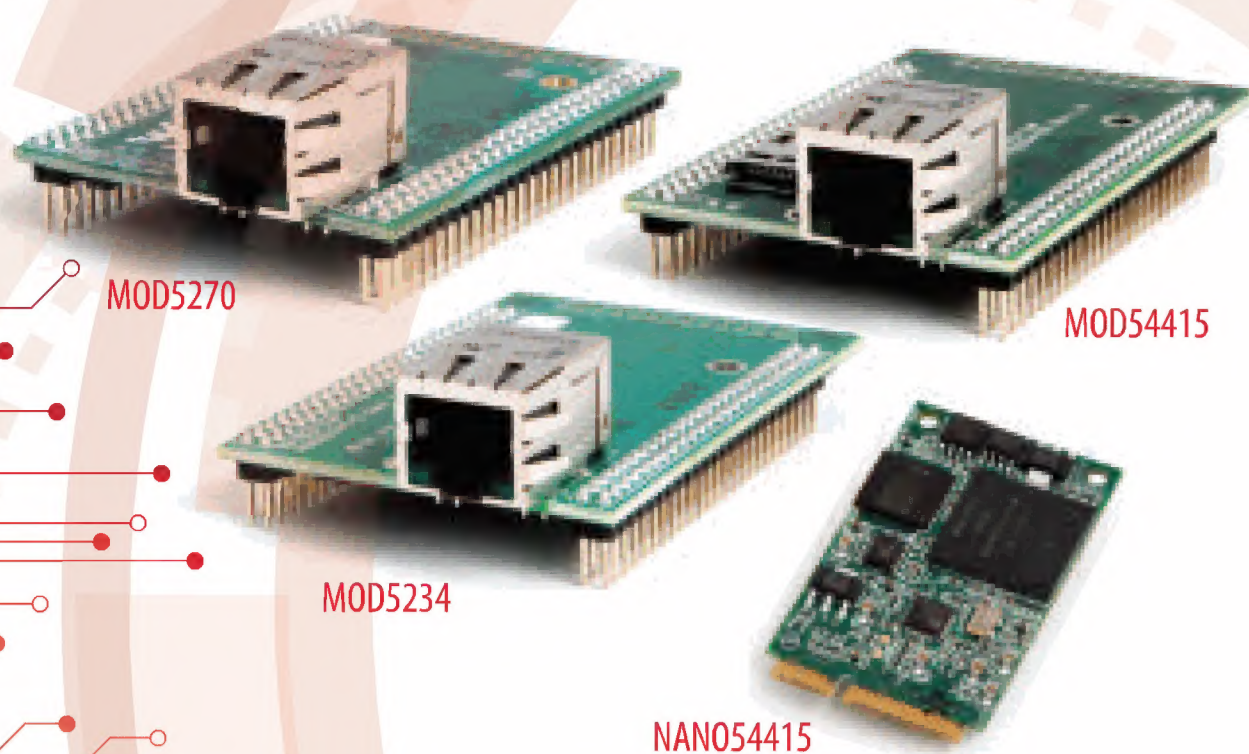
◆ **Let's Get Physical**

Use micro-switches or other physical triggers to manipulate keyboard inputs on your PC.

U.S. \$6.50 CANADA \$7.50



Ethernet Core Modules with High-Performance Connectivity Options

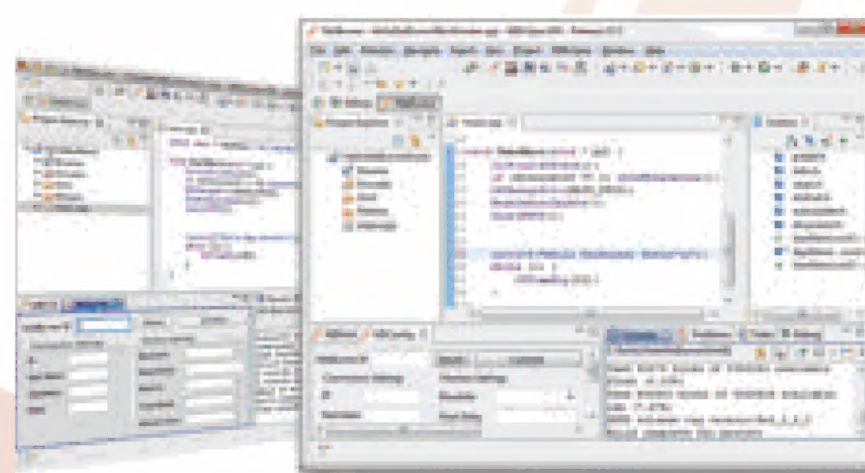


- **MOD5270**
147.5 MHz processor with 512KB Flash & 8MB RAM · 47 GPIO · 3 UARTs · I²C · SPI
- **MOD5234**
147.5 MHz processor with 2MB flash & 8MB RAM · 49 GPIO · 3 UARTs · I²C · SPI · CAN · eTPU (for I/O handling, serial communications, motor/timing/engine control applications)
- **MOD54415**
250 MHz processor with 32MB flash & 64MB RAM · 42 GPIO · 8 UARTs · 5 I²C · 3 SPI · 2 CAN · SSI · 8 ADC · 2 DAC · 8 PWM · 1-Wire® interface
- **NANO54415**
250 MHz processor with 8MB flash & 64MB RAM · 30 GPIO · 8 UARTs · 4 I²C · 3 SPI · 2 CAN · SSI · 6 ADC · 2 DAC · 8 PWM · 1-Wire® interface

Add Ethernet connectivity to an existing product, or use it as your product's core processor



The goal: Control, configure, or monitor a device using Ethernet



The method: Create and deploy applications from your Mac or Windows PC. Get hands-on familiarity with the NetBurner platform by studying, building, and modifying source code examples.



The result: Access device from the Internet or a local area network (LAN)

The NetBurner Ethernet Core Module is a device containing everything needed for design engineers to add network control and to monitor a company's communications assets. For a very low price point, this module solves the problem of network-enabling devices with 10/100 Ethernet, including those requiring digital, analog and serial control.

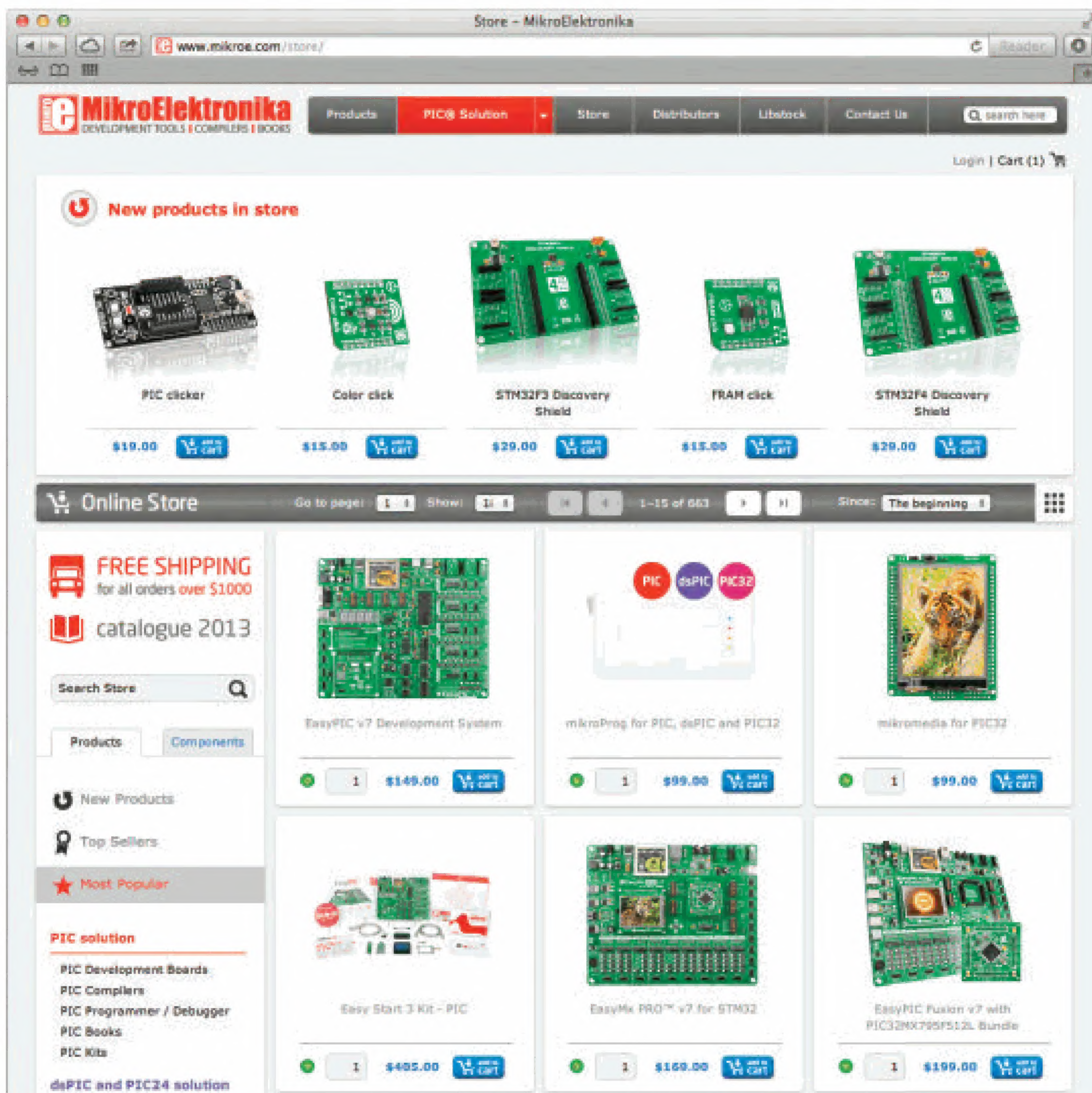
MOD5270-100IR.....\$69 (qty. 100)	NNDK-MOD5270LC-KIT \$99
MOD5234-100IR.....\$99 (qty. 100)	NNDK-MOD5234LC-KIT\$249
MOD54415-100IR.....\$89 (qty. 100)	NNDK-MOD54415LC-KIT\$129
NANO54415-200IR...\$69 (qty. 100)	NNDK-NANO54415-KIT..... \$99

NetBurner Development Kits are available to customize any aspect of operation including web pages, data filtering, or custom network applications. The kits include all the hardware and software you need to build your embedded application.

➤ **For additional information please visit**
<http://www.netburner.com/kits>

the most interesting geek shop of

ELECTRONICS



Turn your ideas into useful gadgets with the help of our development tools. We have everything you need for your project: microcontrollers, development boards, programmers, books, add-on-boards, compilers and a lot more. Put them in your cart, checkout in one easy step, and within a few days you will receive the package at your doorstep. Pay with your credit card or bank transfer.

USE 10% DISCOUNT VOUCHER

At the end of checkout enter voucher code: **NUTS-AND-VOLTS2013**

Offer is valid until 1st of December 2013.

November 2013

28 Quick and Easy USB Keyboard Input

This circuit is perfect for any application where you use a physical trigger to input data onto a computer. Read how it was implemented for video TIC-TAC-TOE and a bag toss game.

■ By William Pippin

34 Build the CHROMATICON

Want to be sure you're playing musical chords correctly? This unit will tell you if you are *and* display a light show in the process.

■ By Matthew Bates

40 Build a Personal Guitar Microphone Pre-Amp

Fine-tune the sound for your guitar just the way you want it with this handy device that will also work with Karaoke machines, boom boxes, and surround sound systems.

■ By Ron Hoffman



46 MakerPlot — The DIY Software Kit Part 2

Besides being a great debugging tool, the real power of this customizable software is in its ability to become a virtual instrument for your microcontroller's data.

■ By John Gavlik and Martin Hebel



Departments

06 DEVELOPING PERSPECTIVES
Sound Off!
07 READER FEEDBACK
24 NEW PRODUCTS
27 SHOWCASE

66 ELECTRO-NET
68 NV WEBSTORE
77 CLASSIFIEDS
78 TECH FORUM
80 AD INDEX

Columns

08 TechKnowledgey 2013

Events, Advances, and News

The first human B2B interface, the world's thinnest keyboard, digital detox, and some tough little fuses are some of the titillating tidbits you'll read about.

12 The Spin Zone

Adventures in Propeller Programming

Digging Into Dynamixels.

Learn the inner workings of these drivers, so you can use them to make your own projects move.

20 Q & A

Reader Questions Answered Here

A Jacob's ladder redux, a video monitor alarm, and robot communication issues are discussed this time.

53 Smiley's Workshop

Programming • Hardware • Projects

Arduino Handheld Prototyper — Part 5.

Data Logging — Take a look at the SRAM and EEPROM on the Arduino, then graduate to SD cards to learn cheap and easy ways to store LOTS of data.

62 Near Space

Approaching the Final Frontier

A New Airframe Design for Near Spacecraft — Part 1.

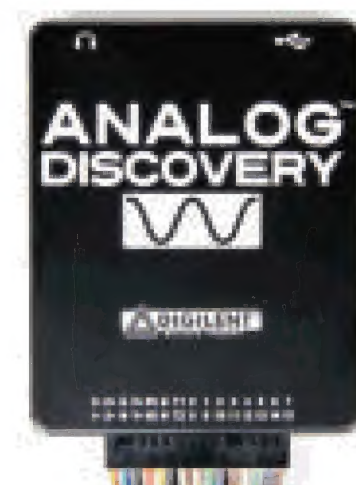
After a recent capsule landing that did not go smoothly, it was the opportune time to come up with a better and stronger design.

71 The Design Cycle

Advanced Techniques for Design Engineers

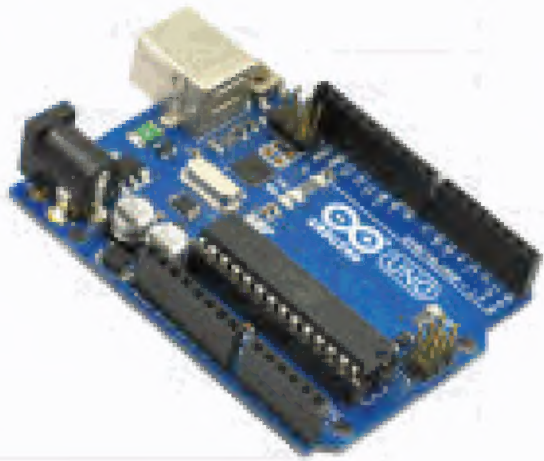
Discovering a Better and More Powerful Engineering Workstation.

One of the first commercial operational amplifier-based computing components was the Model K3-A Adding Component. Unfortunately, you needed a full blown lab to gather the K3's analog results. In contrast, the Analog Discovery packs an oscilloscope, waveform generator, voltmeter, and digital I/O into a space smaller than the first vacuum tube operational amplifier.

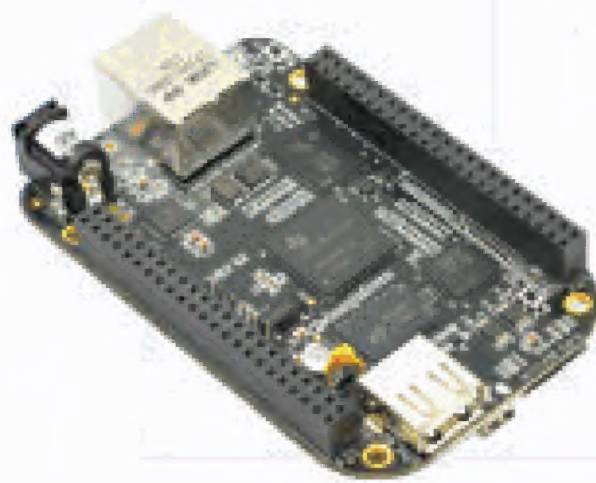


Sense and control, using your...

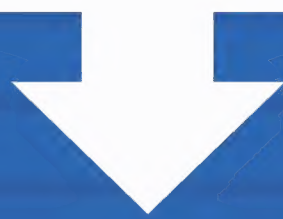
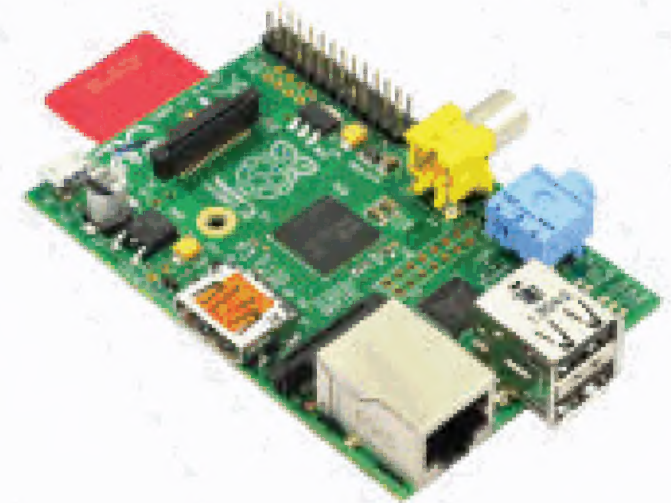
Arduino



BeagleBone

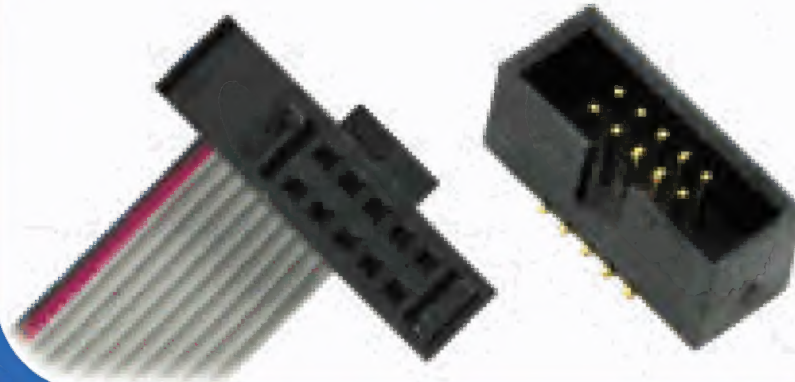


Raspberry Pi



Open Source
software included

a unified cable



100+
options

S
e
n
s
e



Light



Gas



Accelerometer



Gyro



Temp & Humidity

C
o
n
t
r
o
l



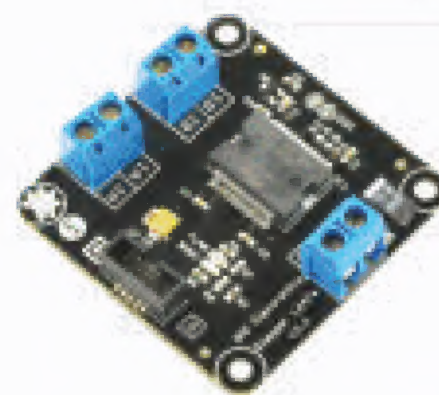
LED



LED Strip



Relay



Motor Control



Color Display

www.GHIElectronics.com/gadgeteering



electronics

10% discount!
NVTENOFF



by Bryan Bergeron, Editor

DEVELOPING PERSPECTIVES

Sound Off!

The ability to add complex sound to a project — be it voice, music, or the chirping of angry birds — has never been so easy or affordable. Given the popularity of personal MP3 players, just about any song or sound imaginable can be downloaded from the Web. Although stand-alone versions of Seri aren't yet available, large vocabulary voice synthesis options are no longer tied to expensive add-ons or desktop computers.

My current favorite for playing sequences of MP3 sounds is the Sparkfun MP3 player shield for the Arduino (\$60), together with their 1.4W class D audio amp (\$8), shown in the accompanying figure. I included the mechanical striker (or hammer) from a 50 year old piano for a size comparison, as well as a reminder of how far we've come in the production and reproduction of sound.

The mechanical striker is quite a piece of craftsmanship — there are multiple layers of felt, chunks of cork, leather, and carefully glued joints. It's a one-of-a-kind, and — together with the wires and other components in an all-mechanical piano — creates a rich tone that can't be fully reproduced on consumer-grade electronics.

The MP3 shield with an accompanying amp, in contrast, are stamped out in the thousands. Furthermore, MP3 files downloaded from the Internet sound the same on every shield. Some of the subtleties may be missing in the audio playback, but most people don't notice.

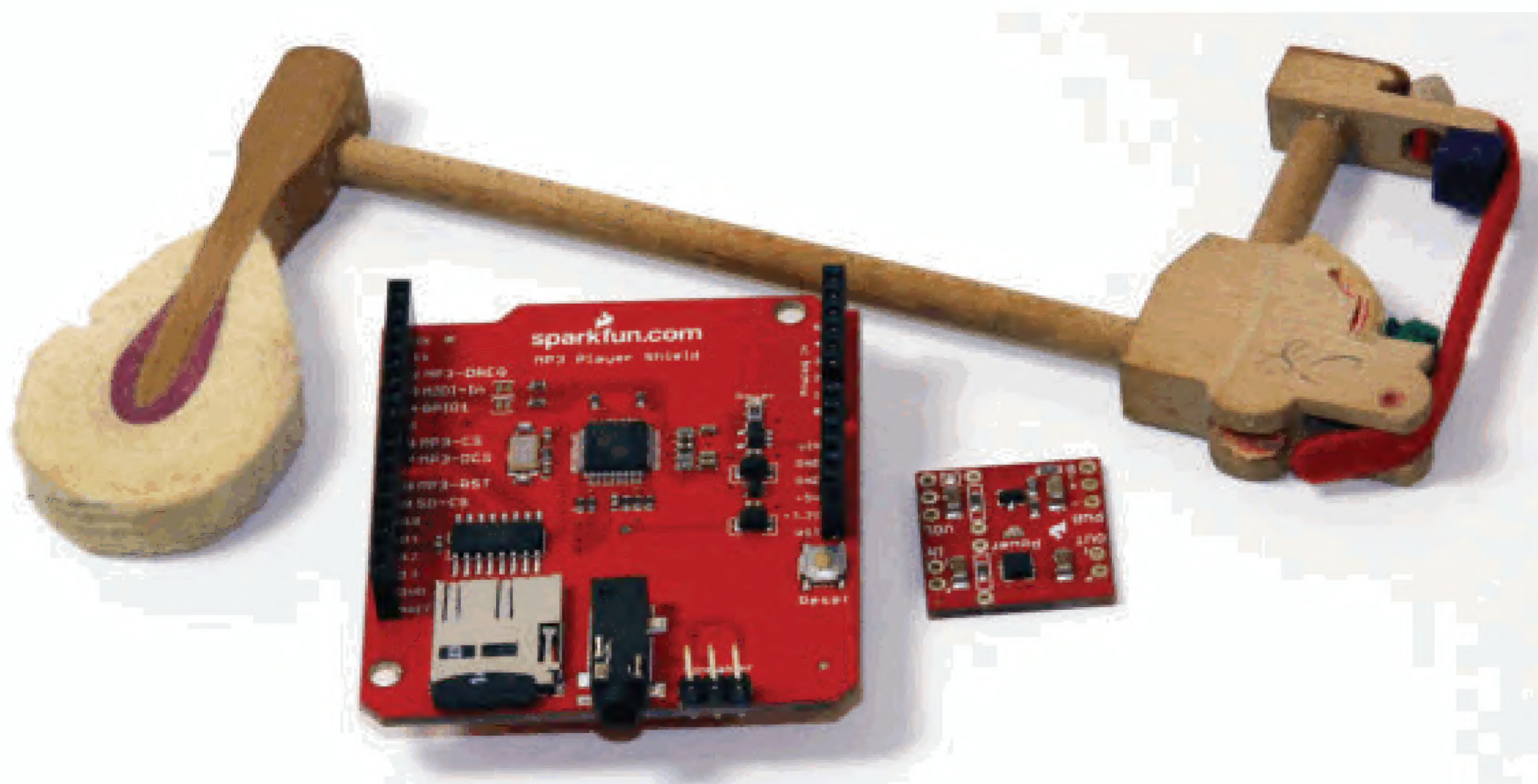
I like the shield as a sound-effects source because of the ease with which MP3 files — stored on the onboard Flash drive — can be accessed and played. For headphones,

it's simply plug-and-play. However, if you need to drive an amp that isn't configured for the low impedance audio output available from the headphone jack, then a pair of the 1.4W audio amps are a necessity (one amp for mono output). It's possible to connect the MP3 player to a different amp, but I'd rather not bother with the capacitors and resistors needed to condition the output of the player.

I've also used the shield/amp combination with a small surface transducer (also available from SparkFun) that can transform a table top or plastic box into a speaker. You'll have to add a resistor to the amp to increase the output of the amp so it drives the surface transducer to full output.

When it comes to voice, it's hard to beat the price/performance of the Emic 2 text-to-speech module from Grand Idea Studio/Parallax (\$60). The Propeller-based board — which is only 1.23" x 1.5" — supports nine voice styles in both English and Spanish. It uses the DECTalk text-to-speech engine to provide dynamic control for pitch, rate, and emphasis.

Last but not least, you should consider the Veho 360 speaker available from Parallax (\$15). The golf ball sized speaker provides 2.4W of output for up to eight hours. It recharges in about four hours through a standard USB cable. Trouble is, I'm going broke giving these things to my nieces, nephews, and coworkers. Not only do they provide a simple, compact output for the SparkFun MP3 player and the Emic 2, but they transform a diminutive iPod into a boom box with amazing bass response. They also double as Christmas tree ornaments — something to consider in December. **NV**



Published Monthly By
T & L Publications, Inc.

430 Princeland Ct.
Corona, CA 92879-1300
(951) 371-8497

FAX **(951) 371-3052**

Webstore orders only **1-800-783-4624**

www.nutsvolts.com

Subscriptions

Toll Free **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277
North Hollywood, CA 91615

FOUNDER

Jack Lemieux

PUBLISHER

Larry Lemieux
publisher@nutsvolts.com

ASSOCIATE PUBLISHER

Robin Lemieux
robin@nutsvolts.com

EDITOR

Bryan Bergeron
techedit-nutsvolts@yahoo.com

CONTRIBUTING EDITORS

Jeff Eckert	Russ Kincaid
Joe Pardue	Fred Eady
Jon McPhalen	Paul Verhage
William Pippin	Matt Bates
Ron Hoffman	John Gavlik
Martin Hebel	

ADVERTISING SALES EXECUTIVE

Richard Lockhart
richard@nutsvolts.com

CIRCULATION DEPARTMENT

subscribe@nutsvolts.com

SHOW COORDINATOR

Audrey Lemieux

MARKETING COORDINATOR

WEBSTORE

Brian Kirkpatrick
sales@nutsvolts.com

WEB CONTENT

Michael Kaudze
website@nutsvolts.com

ADMINISTRATIVE ASSISTANT



Debbie Stauffacher

PRODUCTION

Sean Lemieux

Copyright © 2013 by T & L Publications, Inc.
All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *Nuts & Volts Magazine* assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *Nuts & Volts*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *Nuts & Volts*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: 430 Princeland Court, Corona, CA 92879.

Printed in the USA on SFI & FSC stock.  

READER FEEDBACK

Transistor Twister

In his "Build a Headset Amplifier" article, Ron Anderson explains that experimenters can make several errors in building solid-state circuits.

If an experimenter was to try and make sense of the amplifier schematic, they would immediately get confused to see the schematic symbols of NPN and PNP transistors are indeed reversed.

I like to remember the transistor symbols as: NPN — Never Points iN and PNP — Points iN Permanently

Overall, I did find the design very instructive, and a bit reminiscent of the way we did things in the early 1970s. I always enjoy *Nuts & Volts*!

Robert Trescott
San Diego, CA

In the text of Ron's October article, the transistors were marked incorrectly. These are the proper specs:

Correct:

Transistors Q1-5 2N5401 PNP

Transistors Q6-9 2N5551 NPN

Incorrect:

Transistors Q1-5 2N5551 NPN

Transistors Q6-9 2N5401 PNP

Shedding Light On a Schematic

I enjoyed reading Gerry Shand's article in the June issue about the "Electronic Photocell for Lighting Control." I like his use of the math coprocessor; I had not thought of that before! In the schematic, I have three questions I hope Gerry can help me with:

- Why did you choose the relatively expensive switching voltage regulator instead of a common linear component?

- Why do you employ two resistors in parallel to limit the current through each of your LEDs?

- In case of a power outage, what duration can you expect from your super capacitor until your time chip loses the current time?

Thank you!

Judy May W1ORO
Union, KY

Thank you for your questions, Judy! Here are the answers:

1. I used a switching voltage regulator in lieu of a standard

Continued on page 60



probotix CNC Routers & Control Systems

Imagine What You Can Make or...

Make What You Can Imagine!

Introducing the PBX-BB BeagleBone Powered CNC Controller!

www.probotix.com

The advertisement features a CNC router with a BeagleBone Black board connected to it. A QR code is visible in the bottom left corner.

Post comments on this article at
www.nutsvolts.com/index.php?/magazine/article/november2013_TechKnow13.

ADVANCED TECHNOLOGY

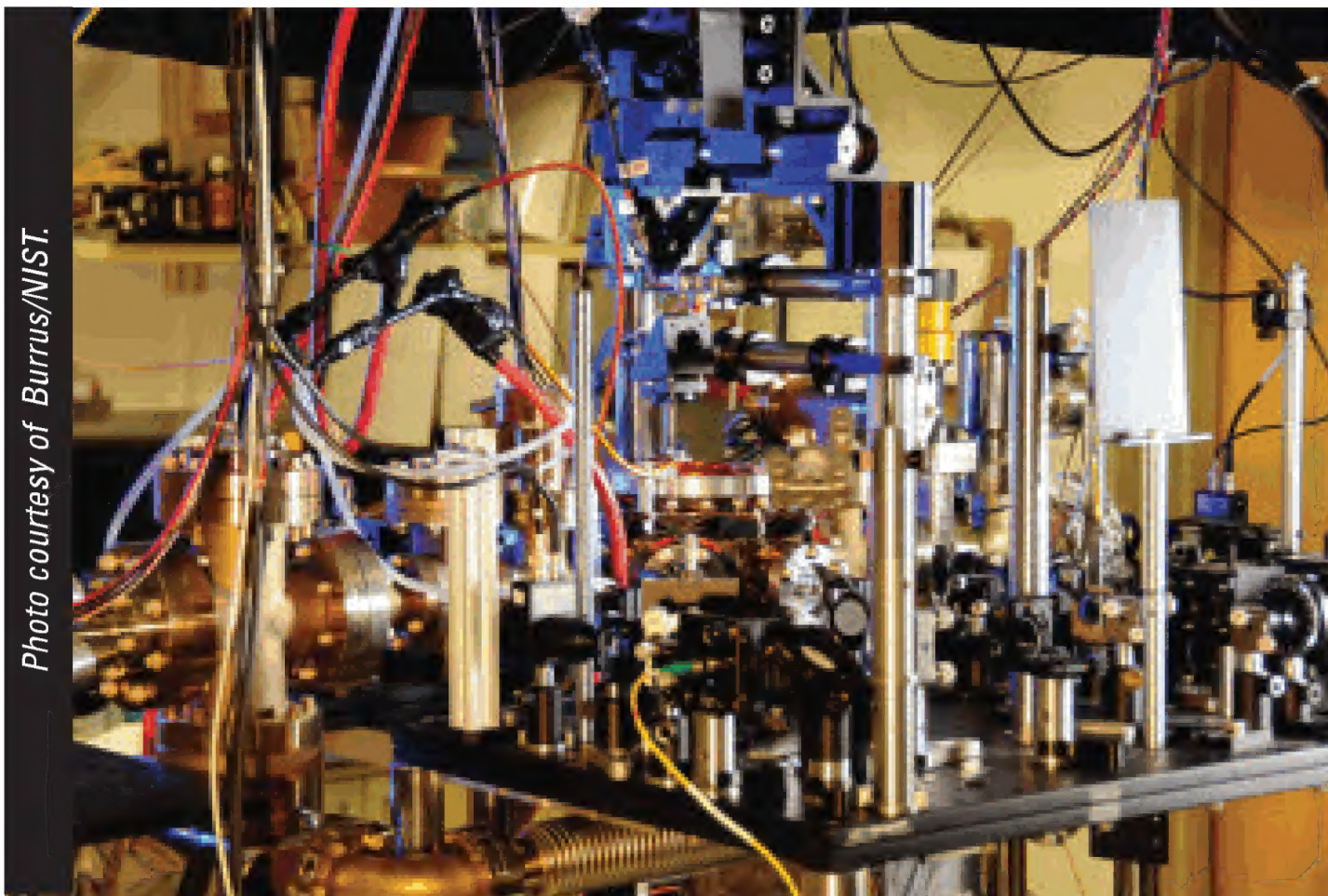


Photo courtesy of Burrus/NIST.

■ NIST's ultra-stable ytterbium lattice atomic clock.

Clocks Set Stability Record

For most of us (time being relative and all that), owning a watch that is accurate to within a few seconds a day is good enough. However, there are some accuracy fanatics out there who are willing to pay big bucks to get super-high-precision timepieces. For example, for about \$7,000 you can pick up a Breitling B-1 which is reported to be accurate to within about three seconds per year.

There are limitations to what you can get out of any quartz movement. If you want maximum precision, you need an atomic clock based on ytterbium atoms like the pair of chronometers recently built by the National Institute of Standards and Technology (NIST, www.nist.gov).

When you get to this level of accuracy, the terminology changes to "stability" which is basically a measure of how precisely the duration of each "tick" matches the others.

The ytterbium ticks are stable to better than two parts in one quintillion (1,000,000,000,000,000,000) which is ten times as good as the crappy old cesium atomic clock the NIST currently uses as the US civilian time standard. (One might logically wonder how NIST can actually verify its stability, but maybe that's why they had to build two of them.)

According to Institute scientists, the new clock "has the potential for significant impacts not only on timekeeping, but also on a broad range of sensors measuring quantities that have tiny effects on the ticking rate of atomic clocks, including gravity, magnetic fields, and temperature."

Just don't expect to wear one on your wrist anytime soon. ▲

First Human B2B Interface

Reaching a new milestone in a technology that might lead to a cornucopia of creepy unintended consequences, researchers at the University of Washington (www.washington.edu) recently achieved what they believe to be the first noninvasive inter-human brain-to-brain (B2B) interface. In an experiment, one researcher was able to send his brain signals over the Internet to create involuntary hand movements in another researcher. (It should be noted that Duke researchers have already made the same connection between two rats, and folks at Harvard linked a rat with a human, but this takes it another step.)

Using "electrical brain recordings and a form of magnetic stimulation," Rajesh Rao remotely operated Andrea Stocco's fingers on a keyboard. "It was both exciting and eerie to watch an imagined action from my brain get translated into actual action by another brain," Rao said.

In the procedure, the sender's brain signals are first recorded and sent to a computer. When the computer detects imaginary hand movements created in the sender's brain, it sends a command over the Internet to a transcranial magnetic stimulation (TMS) machine which fires an electrical coil aimed at the part of the receiver's brain that controls hand movements. This stimulates movement, whether s/he likes it or not.

The planned next step is to figure out how to transmit more complex information from one brain to the other. If that is successful, experiments will begin on a larger pool of subjects. ▲



■ One researcher imagines typing on a keyboard, forcing another to move his fingers.

COMPUTERS and NETWORKING

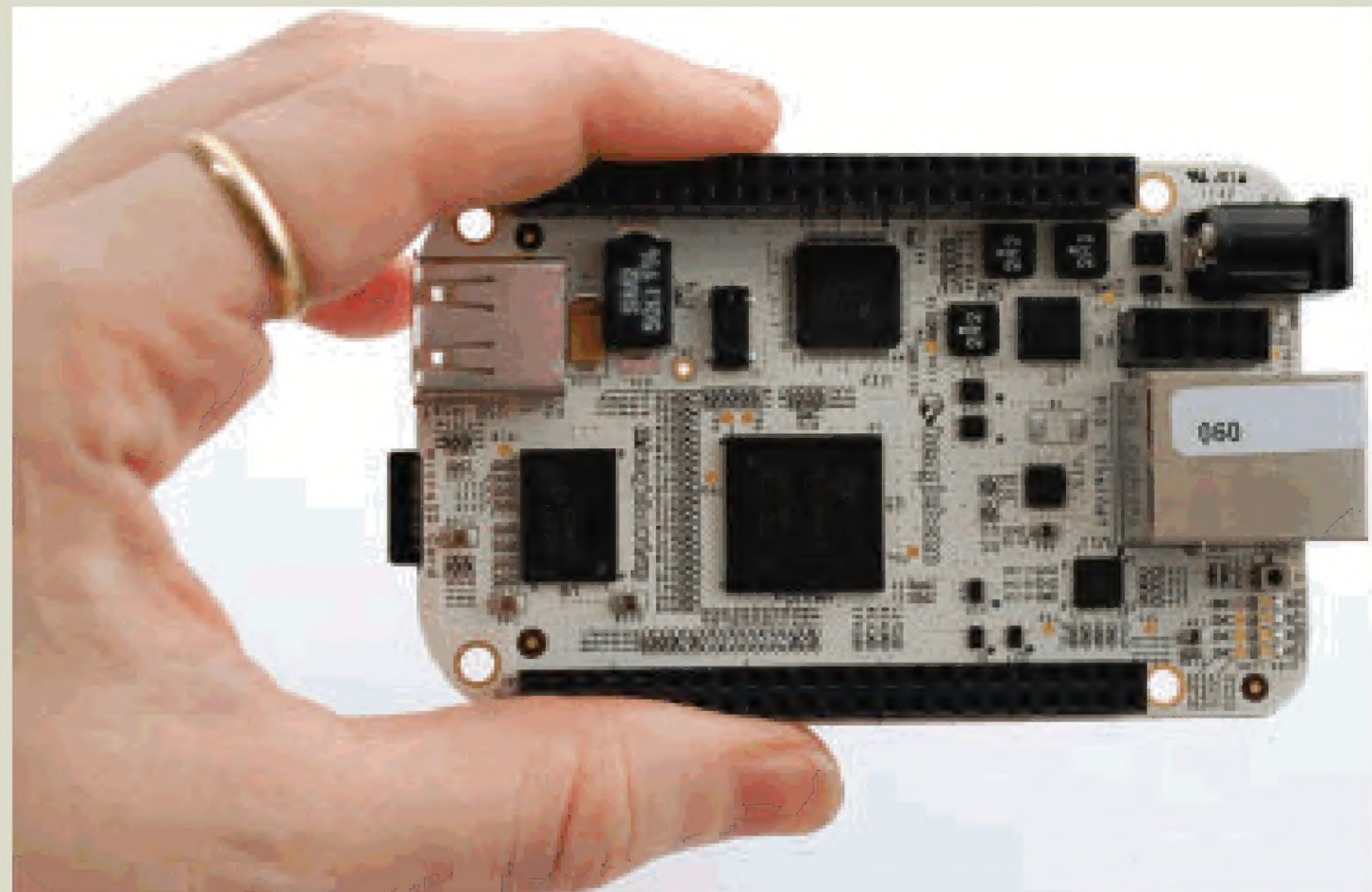
Cheap, Powerful Matchbox

They're not likely to replace their bigger, more powerful counterparts anytime soon, but a slew of "matchbox computers" (a.k.a., thumb PCs) have become available of late, and they are becoming increasingly popular because of their low price, increasing power, and ability to run open source software. One of the latest is the BeagleBone Black which was developed in a joint effort involving Texas Instruments, Digi-Key, and Newark element 14. It is descended from a long line of BeagleBoard products that stretch back to 2008.

Like the better known Raspberry Pi, it is cheap and runs an ARM processor; in this case, a 1 GHz TI Sitara AM3359 ARM Cortex A8. It comes with 512 MB of RAM and 2 GB of storage in the form of an embedded multi-media controller (eMMC). In terms of operating systems, you have a choice of Linux, Android, Ubuntu, and others.

One of the nice features of the BeagleBone is its relatively extensive I/O capabilities, being fitted with 65 GPIO pins (compared to the Raspberry's eight pins). Users also praise its ease of setup.

Using the (included) mini-USB cable, you can hook it up to your PC's power supply and boot it from its own storage. Linux boots in less than 10 seconds, and you can



■ The BeagleBone Black offers 1 GHz performance for the price of a case of dog food.

start working on your project development in less than five minutes. The Bone's main weakness seems to be comparatively less robust audio and video capabilities.

For a mere \$45, its power-to-price ratio is hard to beat, however. For more info, the best place to start is the official user community at beagleboard.org. ▲

World's Thinnest Keyboard

For those of us who grew up with the big fat IBM-style keyboards that offered deep comfortable cupped keys and chattered like a wino's dentures in a blizzard, input



■ The CRS/Atmel/CIT keyboard is less than 0.5 mm thick.

devices have been steadily getting worse. If thinner is better though, then you need to take a look at a new device from CSR (www.csr.com). CSR has incorporated its Bluetooth Smart wireless technology with a wireless touch surface developed by Atmel (www.atmel.com) and Conductive Ink Technology (CIT, www.conductiveinkjet.com) to produce the world's thinnest computing interface.

The device uses Atmel's touch silicon to sense contact points on the surface which consist of a flexible membrane employing CIT printed conductors. The package is less than 0.5 mm (0.02 inches) thick, or roughly the thickness of five sheets of copy paper. You can also draw on it using the nib of a modified pen.

The main idea is to extend the touch interface of iOS7 tablets and smartphones and Windows 8 PCs, but there will no doubt be other applications. As of this writing, you can't actually buy one, but it was formally introduced in September at IFA Berlin (Europe's version of our CES), so commercialization can't be far off. ▲

COMPUTERS and NETWORKING Continued

Digital Detox Offered

The Behavioral Health Services facility at Pennsylvania's Bradford Regional Medical Center (BRMC, www.brmc.com) offers a range of treatment programs for alcohol detox, general psychiatric problems, or a combination thereof, including electroconvulsive therapy. Now however, they also offer treatment for — yes — Internet addiction.

According to BRMC, this is the first hospital-based recovery program in the USA, and it includes the nation's first Digital Device Detoxification Program. According to the center, "Our expert clinicians understand technology related behavioral addictions. They work with individuals, couples, and families to help them better understand and recover from an Internet, video gaming, or technology related behavioral addiction."

To see if you need to check yourself in, just go to the site and answer the Internet Addiction Diagnostic Questionnaire. If you check off five of the eight listed symptoms (e.g., "Do you stay online longer than originally intended?"), you'd better catch the next bus and sign up for the 10 day inpatient program. ▲



■ Are you addicted to Facebook?

CIRCUITS and DEVICES

Curvy OLED TV Now Available

At 55 inches, Samsung's KN55S9C TV is pretty big. Not huge. Not even as big as the 60 inch Vizio you can pick up at Walmart for \$728, but it's curved, which is nice. It is based on organic light-emitting diode (OLED) technology which is good because OLEDs are thinner, lighter, and brighter than standard LEDs. On the other hand, the blue variety has a much shorter lifetime than the red and green ones, so their color will degrade faster. That's why Samsung made the blue subpixels bigger — so you can lower their brightness level to preserve them. OLEDs are more easily damaged by water and, presumably, any beer that you might spill on them during the Super Bowl, which could be a problem.

Aside from the display technology, it's worth noting that the unit has a feature called MultiView, so it can show two programs at once. Two people wearing 3D glasses can watch different high-def programs full screen at the same time. Or, one can watch a 3D program while the other watches one in 2D. The 3D glasses include built-in speakers, so the audio channels won't conflict.

So, does all of this add up to the \$9,000 price tag? It's your call. ▲



■ Samsung's \$9,000 KN55S9C OLED TV.

Stream Your TV

A number of devices can stream TV from your cable or ADSL Internet connection, including the \$99 Apple TV and your Xbox console, but the general consensus seems to be that — at least for now — the Roku device offers the most for the least. The latest model is the Roku 3 which is also the fully loaded model. All four offer 750+ entertainment channels, built-in Wi-Fi, and the ability to play 720p HD video. The Roku 3 adds 1080p video, a remote control with headphone jack, motion control for game playing, and connectivity through dual-band wireless, an Ethernet port, and a USB port. The only apparent drawback to the Roku 3 is that it will not work on standard TV sets — it's HD only. The other three models will operate with virtually any set. The units run from a list price of \$49.99 to \$99.99, but discounts are usually available.

Before you dump the cable or satellite company, you should be aware that the list of free channels consists mostly of obscure sources that you will never watch. Others, e.g., **Disney.com**, provide only snippets of their regular programming, sometimes lasting only a minute or two. There are a couple good cartoon channels for the kids, and if you poke around you can even find some "private" channels like Nowhere TV (courtesy of **thenowhereman.com**) which has a ton of local channels, audio archives, and news and entertainment programming. You can also sign up for Netflix or a flock of other paid movie services. The entertainment industry still won't let you watch your favorite TV shows when they are first broadcast, but for a few bucks you can watch them via Amazon Instant Video the next day. The bottom line is that there's a lot of free and cheap entertainment available over the Internet, but you'll have to sift through quite a bit of chaff to get to the grain. ▲



■ A Roku 3 with remote and headphones.

Tough Little Fuses

There's usually nothing all that interesting about fuses, but the folks at Littelfuse (www.littelfuse.com) recently introduced two series of fuses that are said to be capable of withstanding up to 20 lightning-induced surges without performance degradation. According to the company, they offer better protection than circuit breakers, providing both short-circuit and overload protection for outdoor applications. They also reduce the level of nuisance fuse tripping caused by indirect lightning surges. The 328 series is designed to be installed at the AC input of outdoor luminaire assemblies such as LED lighting used in parking lots, garages, roadways, and outdoor electronic equipment (cameras, displays, etc.). The 688 series (shown in the photo) is a slow-blow ceramic with a breaking capacity of 2,500A @ 70 VDC, and is designed to replace the circuit breakers typically used in telecommunication base transceiver station applications such as power distribution units, remote radio heads, and surge protection device modules. Both are available in cartridge and axial-lead packages. They are available from the company only in quantities of 1,000, though, so if you need fewer than that you'll have to check with your local distributor. ▲



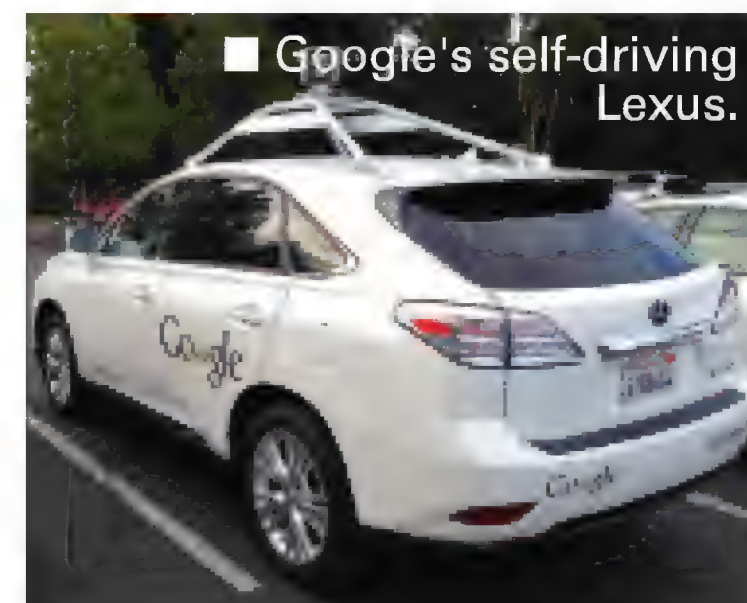
■ Littelfuse series 688 lightning-tolerant fuses.

INDUSTRY and the PROFESSION

Self-Driving Cars: Get Used to It

Autonomous cars may seem pretty exotic at this point, but a new report from Navigant Research (www.navigantresearch.com) provides an examination of the emerging market for advanced driver assistance features leading to semi-autonomous and autonomous driving. The report includes a prediction that self-driving cars will be readily available in the next five years or so and that by 2035 sales figures will reach 95.4 million annually, representing 75 percent of all light-duty vehicles. General Motors is reportedly gearing up to sell a semiautomatic Cadillac in 2015.

Also cited in the report is the Google self-driving Lexus. It notes, "The Google cars use onboard cameras, lasers, radar, and other sensor equipment to monitor road conditions and operate themselves. Proponents say the use of computers and other equipment will make them safer than having humans drive since people sometimes make errors, lose concentration, fall asleep, or drive drunk." So, cheer up. Having one more for the road may not be a problem soon. **NV**



■ Google's self-driving Lexus.

Digging Into Dynamixels

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?/magazine/article/november2013_SpinZone.

As I've stated many times, I love living in Los Angeles. Yes, I know, there are countless wonderful cities around the world, yet this one is my favorite. The beach is close, the mountains are close, the desert is close, and, of course, this is ground zero for entertainment. Two of my friends associated with the entertainment industry — Lou and Matt — expressed an interest in working with Dynamixel actuators. Lou went so far as to send me a couple to play with. The truth is they sat on my desk for a while, but once I dug in, wow! These dudes are really cool.

In the unlikely event that you're not aware of Robotis Dynamixels, they are smart actuators designed to replace standard hobby servos in robotic type applications. They can behave like servos with a 300 degree swing, or — through a software command — be configured to operate as a continuous rotation motor. **Figures 1** and **2** show the top and bottom view of the Dynamixel that I've worked with — the AX-12A. Note that there are two connectors and a spot to write the ID on the bottom; we'll get to this in a second.

What I really like about Dynamixels is that I only need one I/O pin to control a whole bunch of them. The reason is that Dynamixels operate on a single-wire, multi-drop network (**Figure 3**). Communication to the Dynamixel is via half-duplex, 8N1 serial. Even better is that they can talk back; hence, the open-collector data line. We can poll any device on the line for a whole host of useful parameters: position (servo mode only), speed, torque temperature, etc. Trossen Robotics posted an interesting video that demonstrates reading from one set of Dynamixels (with the motors disabled) to set the position in others. In the movie industry, this would be something like a smart one-wire "Waldo." Here's the link: www.youtube.com/watch?v=poPeO7xxA90.

Let me say for the record that I'm not yet an expert at using Dynamixels in practice. What I have done, however, is create what I believe is a very robust driver that allows anyone to access all the features in a very straightforward manner; putting this driver to use is the job of the artist/engineer (*uh ... that would be us*). In this article, I'm going to explain the inner workings of the driver so that you can modify it if you choose, or use it as a starting point for other applications. Of course, I'll show you how to get started so that you can make your own projects move.

Let me also add to the



■ FIGURE 1. Dynamixel top.



■ FIGURE 2. Dynamixel bottom.

record that I'm not the first to do this. When Lou sent the Dynamixels, I checked the Object Exchange (ObEx) and found a driver called DynaComV3 by Dave Ratcliff. It works and got me very excited about using Dynamixels. That said, there were several aspects of Dave's driver that didn't suit my particular programming style. I decided to "liberate" the good ideas that stuck out, and write the rest of my driver from scratch.

Communications Breakdown

Communications between the master controller (Propeller) and the Dynamixel follow a standard command/response model. The master will send an instruction packet that includes the address of the intended target. If that target exists on the buss, it should respond with a status packet. Valid IDs for individual units are 0 (\$00) to 253 (\$FD); ID 245 (\$FE) is used as a broadcast ID for packets intended for all devices.

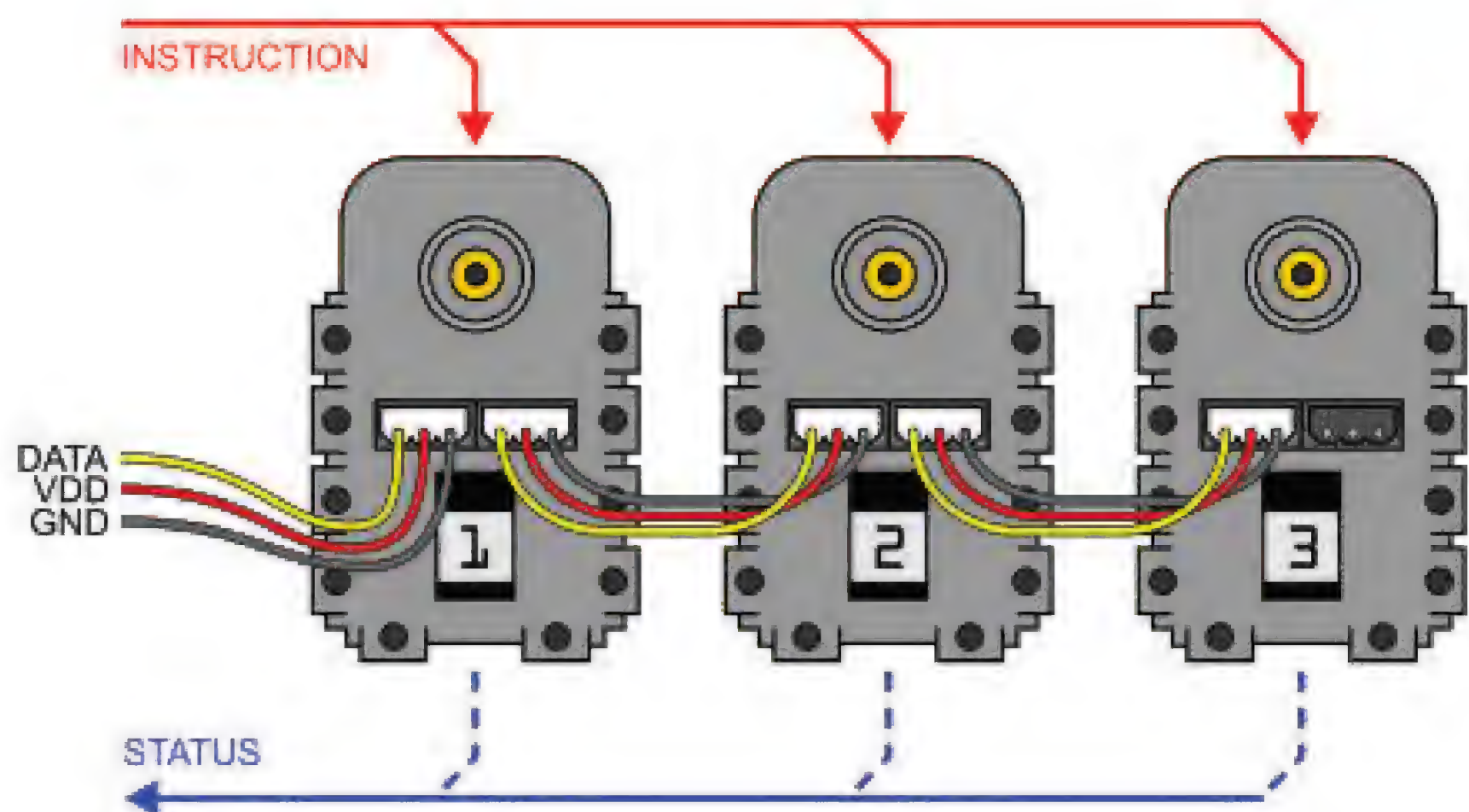
In this case, however, the master does not expect a response. The value 255 (\$FF) is not available as an ID as it is used as part of the packet header. A simple checksum is included in command and response packets to validate the message.

The command (instruction) packet is constructed like this:

```
<$FF>
<$FF>
<ID>
<Length>
<Instruction>
<Parameter1>
...
<ParameterN>
<Checksum>
```

The response (status) packet is similar, with the instruction replaced by an error byte:

```
<$FF>
<$FF>
<ID>
<Length>
<Error>
<Parameter1>
...
<ParameterN>
<Checksum>
```



■ FIGURE 3. Dynamixel network.

The checksum algorithm used is very simple; it's the low byte of the inverted sum of the ID, length, Instruction/Error, and parameter bytes. The Dynamixel object uses this method to calculate the checksum of an array pointed to by *p_buf*:

```
pub checksum(p_buf) | len, cs

    len := byte[p_buf][3] + 1

    cs := 0
    p_buf += 2

    repeat len
        cs += byte[p_buf++]

    return !cs & $FF
```

At the top, we grab the length byte from the buffer and add one to it; we do this to account for the length byte in the message. Then, we clear the working checksum value and skip past the two header bytes. A simple loop accumulates the rest of the bytes in the packet. At the end, we invert the checksum and then trim to eight bits.

By having this method use a pointer to a buffer, we can utilize it to create the checksum for an instruction (command) packet, and use it to calculate the checksum of a status (response) packet. By comparing what we calculate with the checksum contained in the status packet, we can validate the response.

The communications between the Propeller and the Dynamixels are serial which means we'll need a UART to handle the details. We've done this before, and with the Propeller it's quite easy. In this particular case, I've unrolled the transmit and receive code so that it's as fast as possible. The default baud rate for Dynamixels is

1,000,000; pretty zippy, but well within the capability of the Propeller.

What's different about the Dynamixel UART employed in this object is that it doesn't use circular buffers as we'd typically see. There is no need since communications in the system are very well defined: We're going to transmit a structured known-length packet, and optionally receive a structured packet. Instead of managing head and tail pointers, the Dynamixel UART code needs to only know the location of the buffers and packet length values (in the hub). We'll set the packet length for transmitting, and when the UART indicates that it's done we will read the length of the response packet. We do, in fact, need to know the length of the received packet as the last byte is the checksum.

After setting the SIO pin, hub addresses, and baud rate timing values, the UART code drops into a loop that waits for the hub to command a transmission:

```
checkstate    rdlong    axstate, par
              cmps      axstate, M_TX    wz, wc
              if_e      jmp      #transmit
              cmp       axstate, M_TXRX wz, wc
              if_e      jmp      #transmit
              jmp       #checkstate
```

The loop will run until getting a state value of **M_TX** (transmit only) or **M_TXTR** (transmit and then receive). When there is a valid state, the code jumps to **transmit**:

```
transmit      rdlong    txcount, txlenpntr wz
              if_z      wrlong    M_IDLE, par
              if_z      jmp      #checkstate

              mov       txhub, txbufpntr
```

This section reads and validates the length of the transmit packet; assuming all is well, it copies the hub address of the transmit buffer (in *txbufpntr*) to a temporary variable (*txhub*) which is used in the transmit loop. The transmit loop itself is unremarkable; read a byte from the hub, bang it out, increment the working buffer pointer, continue until all bytes are transmitted. As I stated, the code is unrolled for best speed, hence a little long for publication.

At the end of that loop, we need to check if the packet just transmitted expects a response:

```
              cmp       axstate, M_TX    wc, wz
              if_e      wrlong    M_IDLE, par
              if_e      jmp      #checkstate

turnaround    andn      dira, siomask

              mov       t1, tocycles
```

```
:loop        mov       bittimer, cnt
              test      siomask, ina    wc
              if_nc     jmp      #receive
              djnz      t1, #:loop

              wrlong    M_TIMEOUT, par
              jmp       #checkstate
```

At the top, we check the state; if it was transmit only (**M_TX**), we write **M_IDLE** back to the hub to indicate that we're done, and jump back to **checkstate**.

Dropping through, we prepare to receive by setting the SIO pin to input mode. A short loop waits for the start bit of the first byte of the response packet. When detected, the code jumps to **receive** and handles the response. If there is no response, the loop will run through and cause us to write **M_TIMEOUT** to the state variable in the hub. This feature allows us to scan the Dynamixel buss to determine the IDs of any connected devices.

Like the transmit loop, the receive loop is straightforward: after receiving a byte, it stuffs it into the rx buffer, advances the buffer pointer, and then increments the packet length counter. A timeout loop is used to detect the end of the response packet. When the end is detected, the packet length is written to the hub, and the UART state value returned to **M_IDLE**.

It's actually quite easy and useful in other low bandwidth command/response systems (e.g., MODBUS). Now that we have a working UART, let's build up the application code.

By Your Command

The Dynamixel understands seven low-level commands. Most of the time we'll be using the first three. However, the others are there for good reason and the driver allows us to put them to use, as well:

- Ping (\$01)
- Read Data (\$02)
- Write Data (\$03)
- Reg Write (\$04)
- Action (\$05)
- Reset (\$06)
- Sync Write (\$83)

Ping is as simple as its name suggests: It is used to check for the presence of a device on the buss and to retrieve any error bits from it. **Read Data** and **Write Data** are a little more flexible than the names suggest; both allow the transmission of any number of bytes (up to the size of the Dynamixel memory) from or to the actuator. The driver includes a few high-level methods that simplify the use of **Read Data** and **Write Data**.

Reg Write is very much like **Write Data** except the

the values written are buffered until the Action instruction is sent. Using **Reg Write**, we can load several devices with values and then synchronize the acceptance of these values with **Action**.

Reset will, in fact, reset all Dynamixel registers to default values. *It is important to note that this also resets the ID to 1.* This is spelled out in the Dynamixel documentation, but I missed it and spent a frustrating afternoon trying to sort out a programmer-induced communication problem (i.e., both my actuators ended up with the same ID causing response collisions).

Sync Write is a special form of write that allows us to update one or more contiguous registers in multiple actuators. Let's say we want to move four actuators to new targets at speeds suitable for each. We could use **Reg Write** and **Action**, but the setup would require multiple packets. As the Dynamixel designers cleverly placed the target position and speed registers right next to each other, it's easy to update multiple devices with **Sync Write** in a single packet. It is the most involved of the Dynamixel features, and it is fully supported in the driver for those with advanced needs. I'll show you how to use **Sync Write** in a simple application.

Let's start easy with **ping**. Here's the code that implements that feature:

```
pub ping(id) | check

  if ((id < 0) or (id > $FD))
    return E_ID

  txbuf[0] := $FF
  txbuf[1] := $FF
  txbuf[2] := id
  txbuf[3] := $02
  txbuf[4] := INST_PING
  txbuf[5] := checksum(@txbuf)

  tx_execute(6, M_TXRX)

  if (state == M_IDLE)
    if (rxlen == 6)
      if (rxbuf[2] == id)
        if (checksum(@rxbuf) == rxbuf[5])
          rxerror := rxbuf[4]
          return id

  return E_ID
```

Ping expects a response, so the first thing we check for is a valid device address (0 to 253). The six-byte message is built per the Dynamixel protocol description with the checksum in byte five of the buffer. The **tx_execute** method activates transmission by the UART cog:

```
pub tx_execute(len, mode)

  repeat
    while (state > M_IDLE)

  rx_clear
  txlen := len
  state := mode

  if (mode == M_TXRX)
    repeat
      until (state =< M_IDLE)
```

While not likely, it is possible that a programmer could attempt to send back-to-back packets, so this code will wait if the UART is busy. Once it's free, the receive buffer and length are reset (with **rx_clear**), the length of the transmit buffer is moved into **txlen**, and the mode (transmit or transmit/receive) is moved into **state** (which is the variable being monitored by the UART cog). If the transmitted packet is expecting a response, the program will wait until **state** is changed to **M_IDLE** or **M_TIMEOUT**.

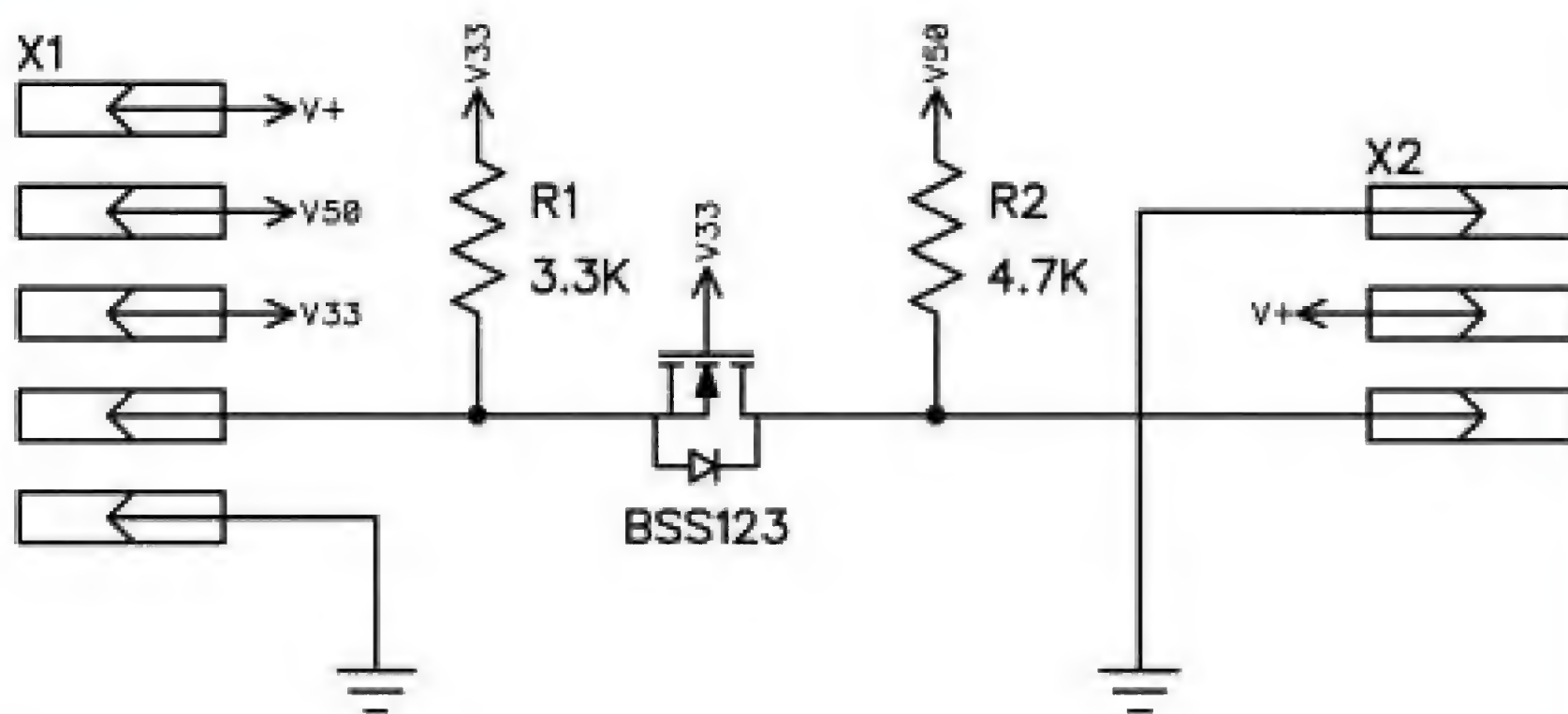
Getting back to the **ping** code, let's assume the device pinged did respond (no timeout). The first check passes. Is the length of the response correct (6)? Does the response include the correct ID? Does the checksum in the packet match what we calculate from the packet? If the answer to each of these questions is yes, then we'll return the ID of the device to indicate a good ping. Otherwise, we'll return an error code.

Error codes are negative since values in the system are always positive. **Ping** also returns an error byte which is saved to an object variable called **rxerror**, and made available to the object's parent through the **rx_error** method.

Okay, I know this stuff is a bit gnarly, so let's build a little program to see things work. Load up **jm_dynamixel_scanner.spin** at the article link and give it a go. You'll need a bit of an interface between the Dynamixel and the Propeller. My preferred interface is the circuit shown in **Figure 4**. That said, a 10K pull-up to 5V

BILL OF MATERIALS

Q1	BSS123	Mouser 512-BSS123
R1	3.3K	Mouser 71-CRCW0805-3.3K-E3
R2	4.7K	Mouser 71-CRCW0805-4.7K-E3
X1	0.1 Male	Mouser 517-6111TG
X2-Jack	SPOX 3-pin	Mouser 538-22-03-5035
X2-Plug	SPOX 3-pin	Mouser 538-50-37-5033
X2-Socket	Female	Mouser 538-08-70-1040
PCB		DipTrace/Bay Area Circuits

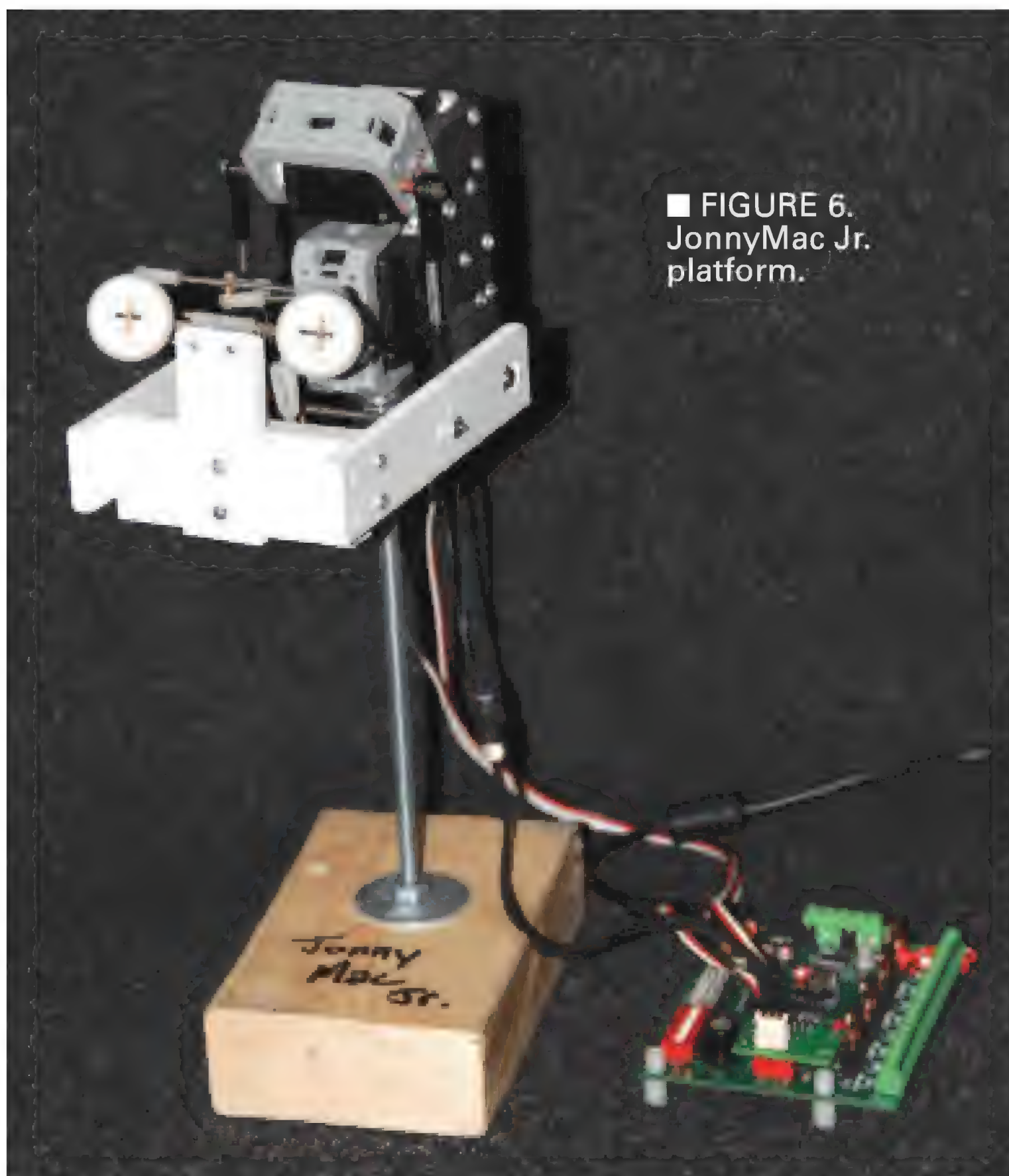


■ FIGURE 4. Dynamixel interface schematic.

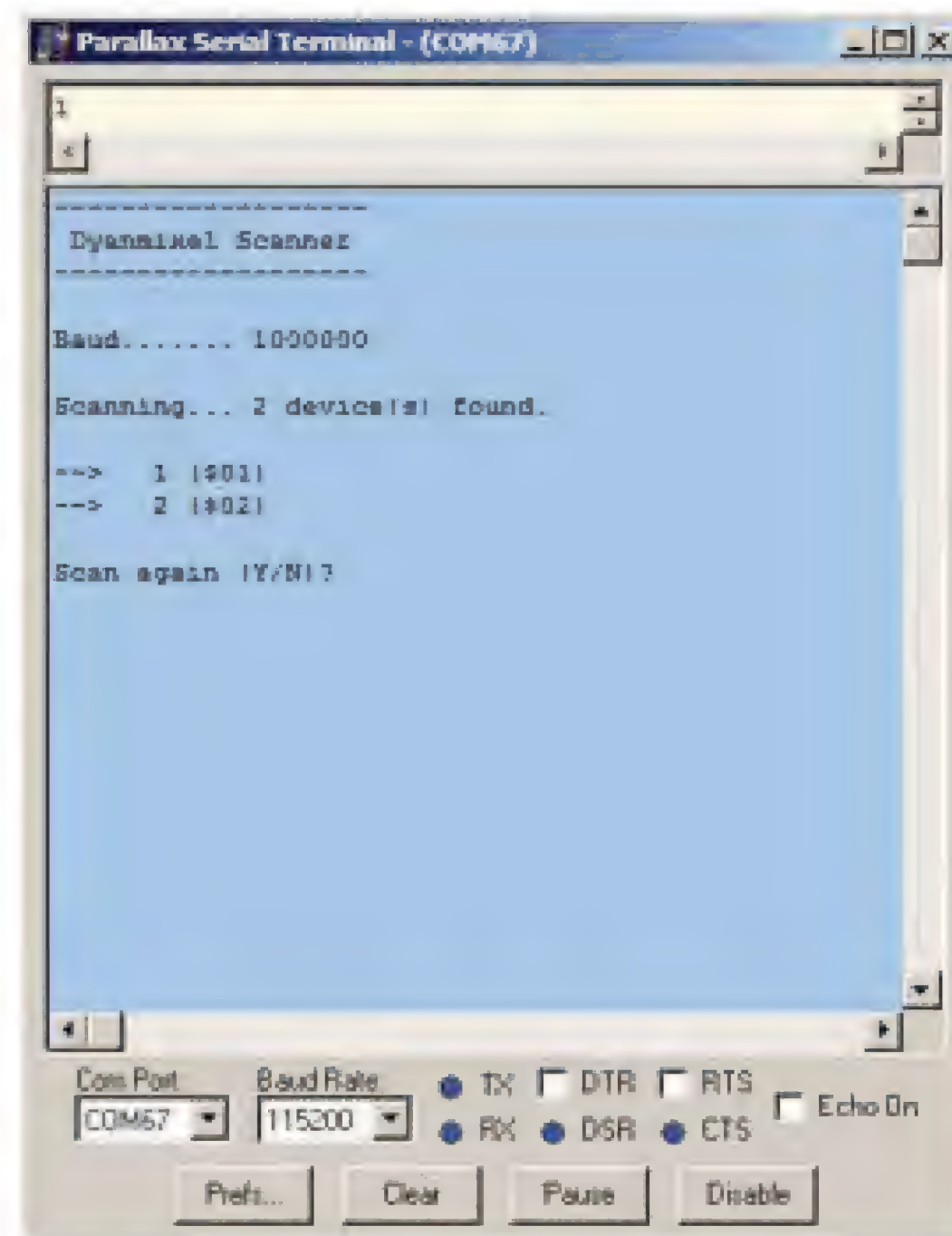
with 1K between the Propeller pin and the pull-up will work, too (see the object code for the alternate connection).

After downloading the program, open a terminal window and then press a key. You'll see a menu for setting the baud rate for the driver — press [1] for 1,000,000 baud (unless you know your actuators have been set to a different rate). After that, the code uses a method called **scan** to document any devices connected to the Propeller.

In the middle of the **scan** method, you'll find this loop which uses the driver's **ping** method:



■ FIGURE 6. JonnyMac Jr. platform.



■ FIGURE 5. Dynamixel scanner report.

```
repeat id from $00 to $FD
  cursor_xy(12, 6)
  show_id(id)
  if (dy.ping(id) == id)
    cursor_xy(0, devices+8)
    term.str(string("--> "))
    show_id(id)
    devices += 1
  pause(50)
```

This code loops through all possible IDs, pings each, and if there is a successful response it is noted on the output. When complete, the number of devices detected is displayed. **Figure 5** shows the result of the program running on my test setup.

Note that there is a 50 ms delay between packets. While talking with Kyle at Trossen Robotics, I got the impression that Dynamixels — even with their crazy-fast default baud rate — don't want to deal with more than about 30 packets per second. While I cannot find any documentation to support this, the guys at Trossen have a lot of experience with the Dynamixels and have been incredibly helpful when I've called. This helps explain the use of **Sync Write**; while a bit fussy to set up, it allows one to update registers in several actuators using a single packet.

While **ping** is useful to inventory devices connected to a system, that's about where its utility ends. To do useful things with the Dynamixel, we need to read and write registers. The driver includes low-level methods called **read_data** and **write_data**.

These methods are flexible enough to work with one byte up to all 50 in the Dynamixel memory.

To simplify things for practical use, I created shell methods: **rd_byte**, **rd_word**, **wr_byte**, and **wr_word**. These make it very easy to read or write bytes or words (the only data types in the Dynamixel memory).

Let's say we want to read the temperature from device #1. It's easy with the **rd_byte** method:

```
temp := dy.rd_byte(1,  
dy#P_PRESENT_TEMPERATURE)
```

The first parameter is the device ID; the second is the register to read. The Dynamixel documentation includes a demo program (in C); I lifted the register name constants from it. By using **rd_byte**, **rd_word**, **wr_byte**, and **wr_word** with the register constants, we don't have to create method calls for every register that we'd want to access. Yet the code is still quite readable.

It's always more fun to play when you have something real to work with, so Matt built a prototype platform; you can see it in **Figure 6** (note that he *smarmily* labeled it "JonnyMac Jr."). It's a pan/tilt mechanism structured as a head, and even includes a set of moving eyes (that use micro-servos; sadly, there

are no Dynamixels that small). I connected the Dynamixels to an EFX-TEK HC-8+ using a little adapter that is a variation of the schematic and printed circuit board (PCB) files I've included with this project. The test program I'm running on the rig is called **jm_dynamixel_pantilt.spin**.

In the course of testing communications, I decided I'd try to read the entire contents of an actuator with one large packet instead of reading individual registers with several small packets and the requisite delays between them. It works well, and demonstrates the flexibility of the **read_data** method.

For example, to read the entire contents of the PAN actuator (#1), we would do this:

```
check := dy.read_data  
(PAN, 0, 50, @regbuf)
```

The first parameter used is the device ID; the next is the starting address in the actuator to read; the third parameter is the total number of bytes to read; and, finally, the last is the address of the array to hold the bytes read. In this case, we've defined a byte array called **regbuf** that will hold the values pulled from the actuator. As with **ping**, a good read will return the ID of the actuator; a negative value indicates an error in the transaction.

The Dynamixel memory is a mix of byte and word registers, so I created a couple methods to read bytes and words from a byte array. One of the registers I wanted to display in my test project is the voltage in the actuator. This is a single-byte value expressed in tenths of a volt. This bit of code retrieves the value and displays it properly:

```
value := dy.get_byte  
(@regbuf, dy#P_PRESENT_  
VOLTAGE)  
rjdec_xyw(value/10, col,  
7, 6)  
term.tx(".")  
term.tx(value//10 + "0")
```

RESOURCES

Jon "JonnyMac" McPhalen
jon@jonmcpalen.com

Parallax, Inc.
www.parallax.com
Propeller chips and
programming tools

Trossen Robotics
www.trossenrobotics.com
Dynamixel actuators, accessories,
and robotics parts

EFX-TEK
www.efx-tek.com
Controllers for prop and
display builders



HEXBRIGHT

HEXBRIGHT

The
Brightest
Most Durable
Arduino
Flashlight.
Period.

hexbright.com

The output of the test program is shown in **Figure 7**; as you can see, I'm displaying several registers from the pan and tilt actuators, including the present position which I can modify by moving the rig. This is possible because I disabled the torque of both (with **Sync Write**) before dropping into the report loop.

In Sync

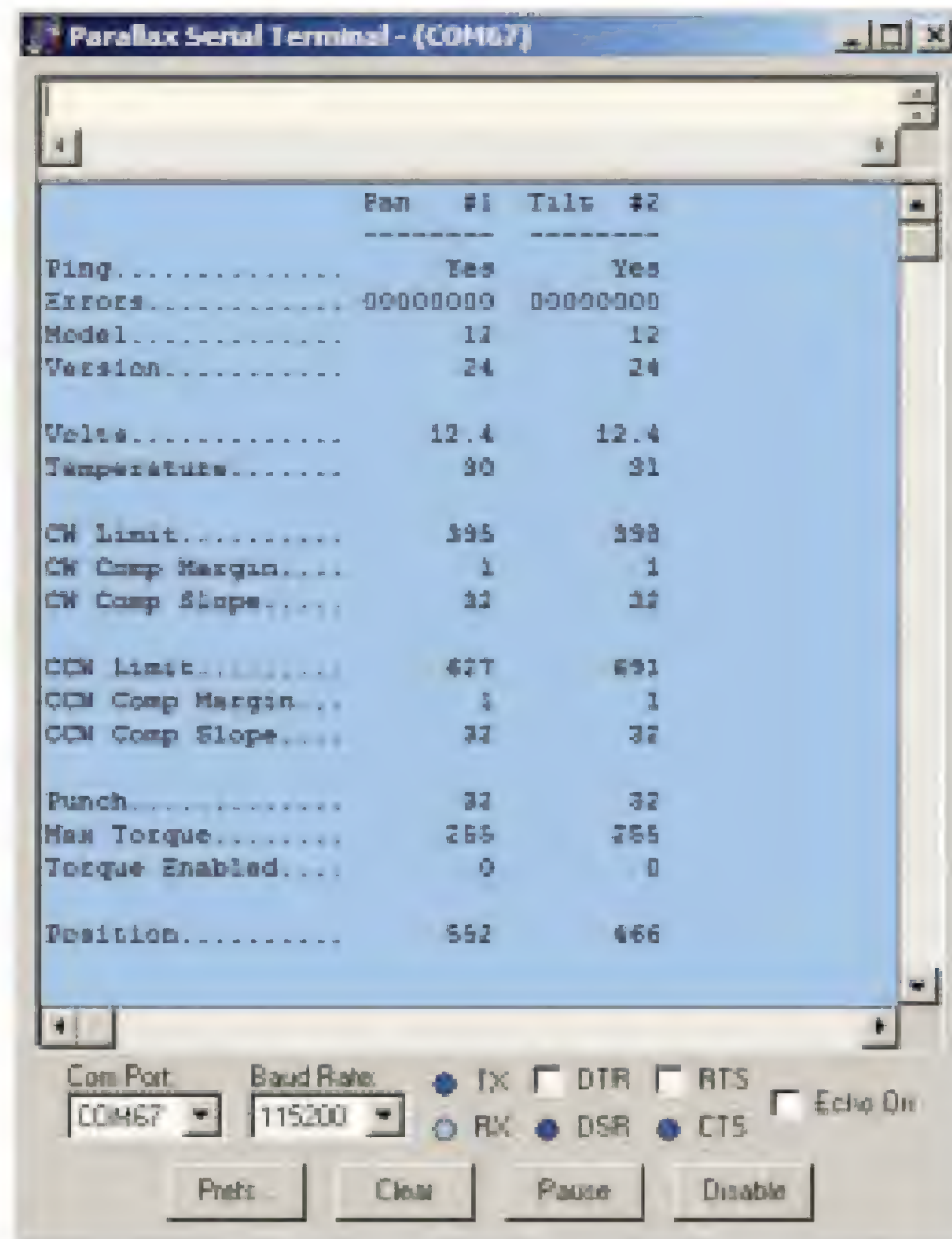
I mentioned earlier that the **Sync Write** instruction is the trickiest aspect of using Dynamixels, so I've saved it for last. There's really no two ways about it. We're forced to build the **Sync Write** packet manually, and then we call the method. Let me show you a bit of working code which will help me explain things. This is the code I use to disable the actuators in the pan/tilt rig:

```
dy.tx_clear
dy.tx_putbyte( 7, PAN)
dy.tx_putbyte( 8, 0)
dy.tx_putbyte( 9, TILT)
dy.tx_putbyte(10, 0)
dy.sync_write(2, dy#P_TORQUE_ENABLE, 1)
```

We start by clearing the transmit buffer. In the structure of the **Sync Write** packet, values are written to that buffer starting at index 7. For each of the actuators affected, we need to load the ID and any parameter bytes. The **tx_putbyte** and **tx_putword** methods are used to move values into the transmit buffer. The final line in the example above calls **sync_write** declaring that two actuators will be updated, the starting register in each is the torque enable byte, and that we're providing just one byte for each actuator.

In this example, keeping track of the registers is very simple. In applications where several registers are involved — especially of mixed types — the **tx_putbyte** and **tx_putword** methods have an optional behavior. To move the pan and tilt actuators to their centered position at a moderately slow speed, we could do this:

```
dy.tx_clear
dy.tx_putbyte( 7, PAN)
dy.tx_putword(-1, 511)
dy.tx_putword(-1, 32)
dy.tx_putbyte(-1, TILT)
```



■ FIGURE 7. Dynamixel pan-tilt report.

```
dy.tx_putword(-1, 511)
dy.tx_putword(-1, 32)
dy.sync_write( 2, dy#P_
GOAL_POSITION_L, 4)
```

Note the use of -1 in the index position of **tx_putbyte** and **tx_putword**. Here's the story: When we specify an index of zero or greater, we reset an internal index variable. That index is advanced by one for **tx_putbyte** and by two for **tx_putword**. By specifying -1, we don't have to keep track of registers manually. We'll always have to use index 7 for the first register; the rest can be automated with -1.

Get Your Motors Runnin'

Okay, you know the drill. Connect your Dynamixels and let 'em rip. I feel pretty confident in the

driver but, as ever, you should check ObEx from time to time for updates. Matt, Lou, and I have lots of neat ideas for Dynamixels, and now that we have a driver we can build them.

What will you build? For you robot fans, check out **Figure 8** — a platform built by the guys at Trossen that uses 12 AX-12 actuators. I don't do a lot of work with robotics, but it seems to me that the Propeller is an excellent processor for those applications. With this driver, a few Dynamixels, and a copy of the Dynamixel documentation, you're ready to roll. Or walk. Or crawl. It's your choice! Until next time, keep spinning and winning with the Propeller! **NV**



■ FIGURE 8. Trossen robot.

ITEM #2134

\$7⁹⁵

(qty. 5)



- DRV8834 driver
- 4-layer PCB
- Low voltage (2.5 V - 10.8 V)

ITEM #2133

\$11⁹⁵

(qty. 5)



- DRV8825 driver
- 1/32 microstepping
- 4-layer PCB

Stepper Motor Drivers

ITEM #2128

\$9⁹⁵

(qty. 5)



- A4988 driver
- 4-layer PCB

ACS711EX Current Sensor Carrier

ITEM #2453

\$3⁹⁵

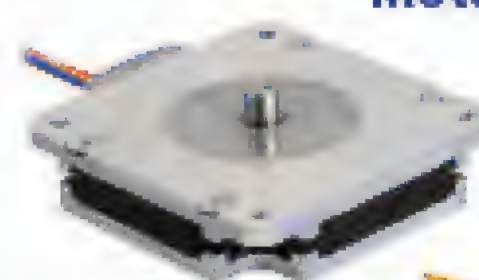


- Hall effect-based current sensing
- Low resistance current path
- Electrical isolation up to 100 V
- Available with ± 15.5 A or ± 31 A range

Sanyo Pancake Stepper Motor 50x11mm

ITEM #2297

\$49⁹⁵



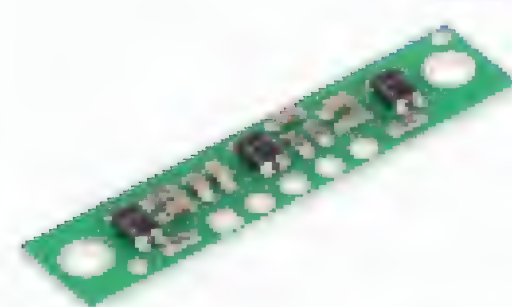
- Flat profile
- 1 A/phase @ 4.5 V
- 14 oz-in

Other shapes and sizes available

QTR-3RC Reflectance Sensor Array

ITEM #2457

\$4⁹⁵



- Great for line-following robots
- Light weight: 0.6 g
- Small size: 1.25" x 0.3"

NiMH Battery Packs and iMAX-B6AC Balance Charger and Discharger

STARTING AT

\$4⁴⁹

ITEM #2260

\$49⁹⁵



Single cells and battery holders also available



Addressable RGB 30-LED Strip, 1 m

ITEM #2543

\$34⁹⁵



Waterproof, individually addressable RGB LED strip that runs on 5 V. Can be chained to form longer strips or cut for shorter sections. Also available in 2 m and 5 m lengths.

Universal Aluminum Mounting Hub Pair

ITEM #1203

\$7⁴⁹/pair



Other shaft sizes available

Mount custom wheels and mechanisms to 5 mm motor shafts.

Pololu Wheels

STARTING AT

\$4⁹⁸/pair



Dual MC33926 Motor Driver Shield for Arduino

ITEM #2503

\$29⁹⁵



Control two brushed DC motors with an Arduino or Arduino-compatible board. Operates between 5 V and 28 V and can deliver a continuous 3 A per motor.

Optical Encoder Pair Kit for Micro Metal Gearmotors

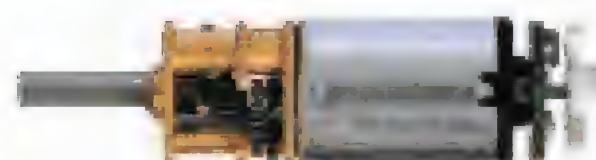
ITEM #2590

\$8⁹⁵



Compatible Gearmotors

\$16⁹⁵



Extends only 5 mm past motor

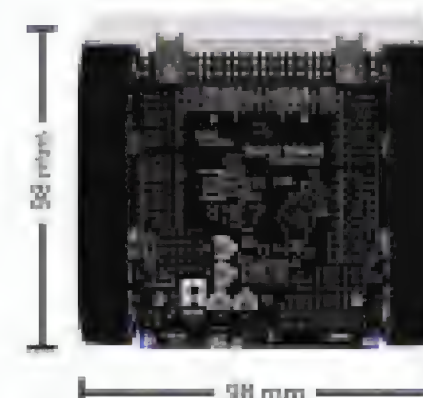
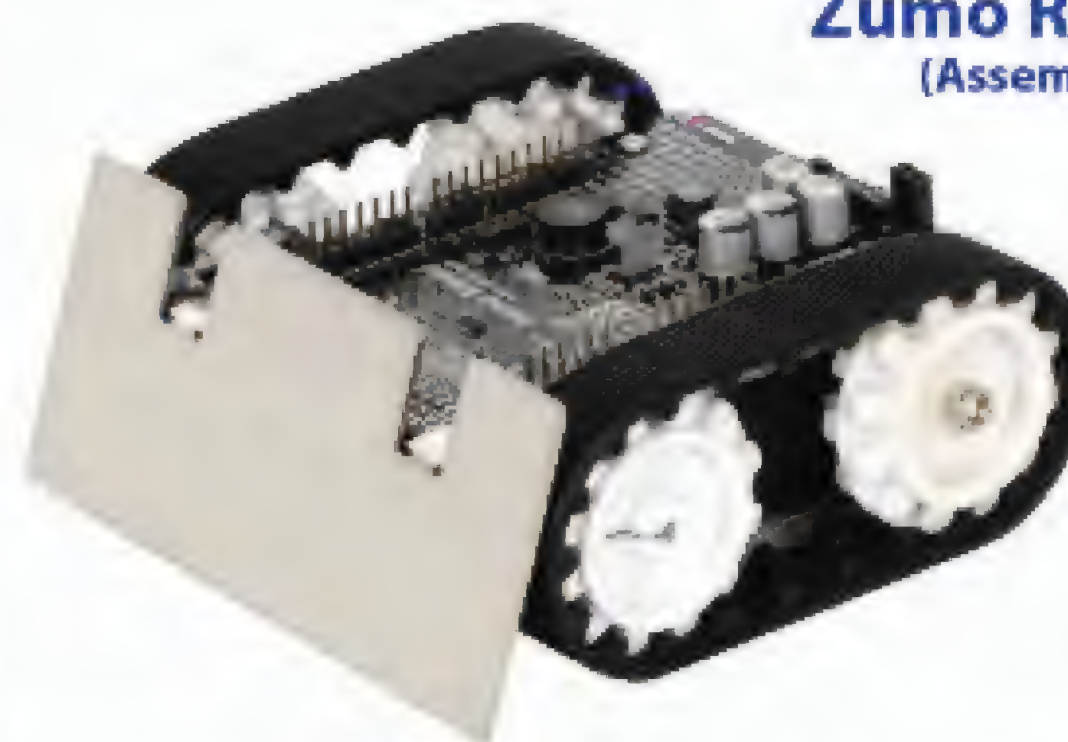
Add quadrature encoding to your micro metal gearmotors!

- 3- and 5-tooth wheels included
- 3.3 V and 5 V versions available
- Works with our growing assortment of gearmotors with extended back shaft (gearmotors sold separately)

Zumo Robot for Arduino (Assembled with 75:1 HP Motors)

ITEM #2506

\$99⁹⁵



Arduino-controllable tracked robot small enough for mini-sumo (less than 10 cm x 10 cm) and flexible enough for you to make it your own.

Includes:

- Two micro metal gearmotors coupled to a pair of silicone tracks
- Stainless steel bulldozer-style blade
- Array of six infrared reflectance sensors for line following or edge detection
- 3-axis accelerometer and magnetometer
- Buzzer for simple sounds and music

Individual parts and kit version also available — build your own configuration!

Build and customize your Zumo!

In this column, Russ answers questions about all aspects of electronics, including computer hardware, software, circuits, electronic theory, troubleshooting, and anything else of interest to the hobbyist. Feel free to participate with your questions, comments, or suggestions. **Send all questions and comments to: Q&A@nutsvolts.com**

- **Jacob's Ladder Redux**
- **Video Monitor Alarm**
- **Robot Communication**

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?/magazine/article/november2013_Q&A.

More On the Jacob's Ladder

Q I have nothing but good news! First, I want to thank Mr. Carlson for his response and advice; slowing the frequency down made all the difference in the world. Second, I would like to thank you! Without your help, Russ, I know I would not have had this much success.

I have not changed the last revision of the circuit I previously submitted. I reengineered the construction of the electrodes making them easier to adjust, and fine-tuned the oscillator. Now, I'm getting a spark to gap 1/8" and climb

a 10" set of electrodes ending at a final gap of 7/8".

There are two coils that I have been working with — both 12V. The first is a smaller twin cylinder type coil; the second is an old can type automobile coil. In my opinion, the smaller coil actually does a better job. It gives a longer more active spark, but it is not as bright.

I want to build a switch circuit. Push the button and it will turn on for 15-20 seconds or whatever, and then turn off. I know I can do this but I have a concern. Additional circuitry will undoubtedly consume more power.

My thoughts are to upgrade to another power supply. Another 12V

with slightly higher amperage: 13.8V at 5A or 15V at 8.4A.

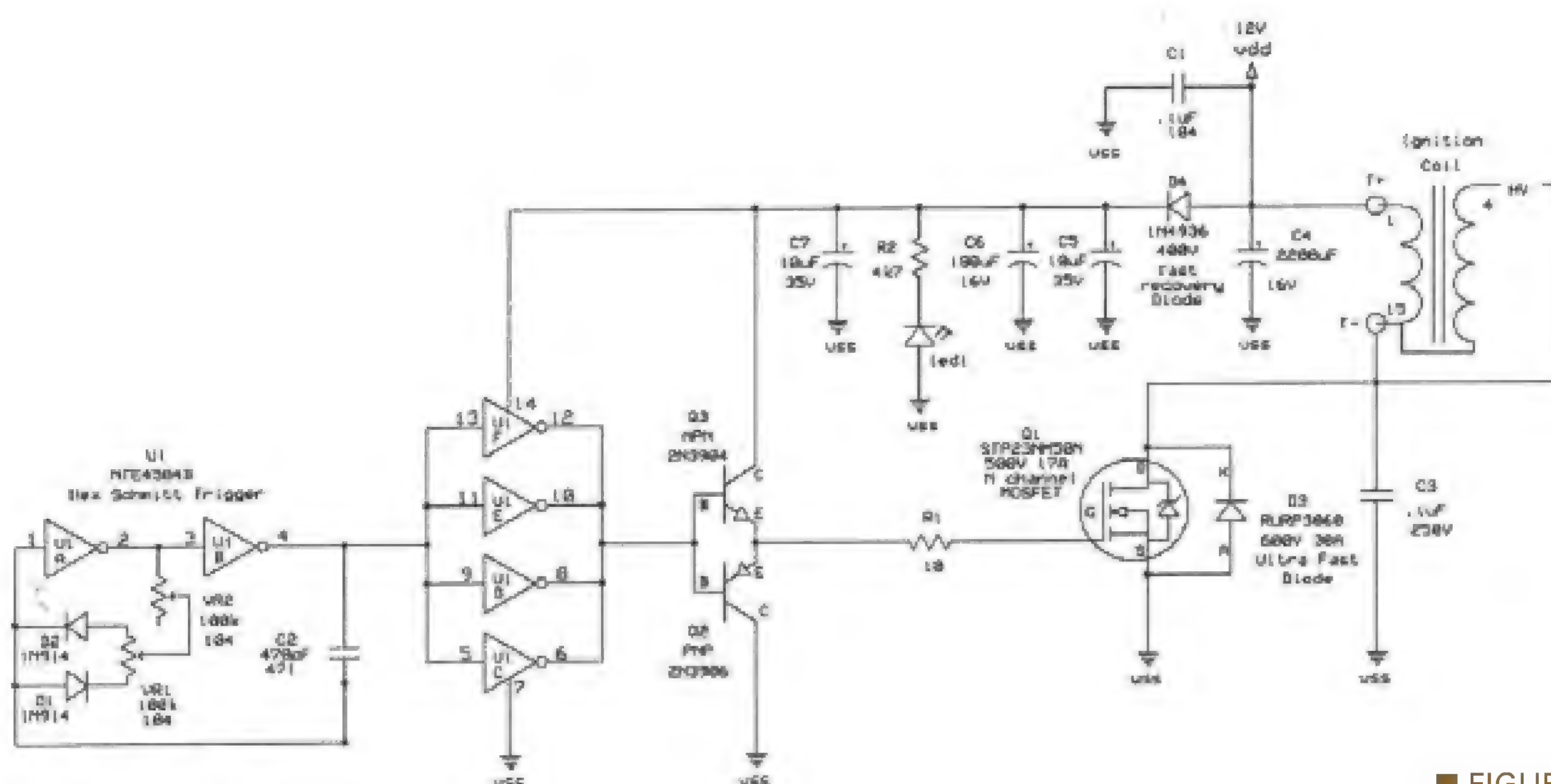
I'm not sure if more voltage, more amperage, or if both are the way to go. I don't want to overdo it.

As the circuit stands now, from a 12V 5A supply it delivers 9.8 volts to the coil — a 2.2 volt loss.

I do not know how to check the current consumption on this circuit. There are several stats I would like to check, but I need to read up on how.

The hex Schmitt trigger has an 18 volt max voltage limit, and the two aluminum electrolytic capacitors are rated at 16V. The other components exceed these voltage limits.

— Mike



■ FIGURE 1.

A I have reproduced your circuit here (which was in the September issue) for clarity in **Figure 1**.

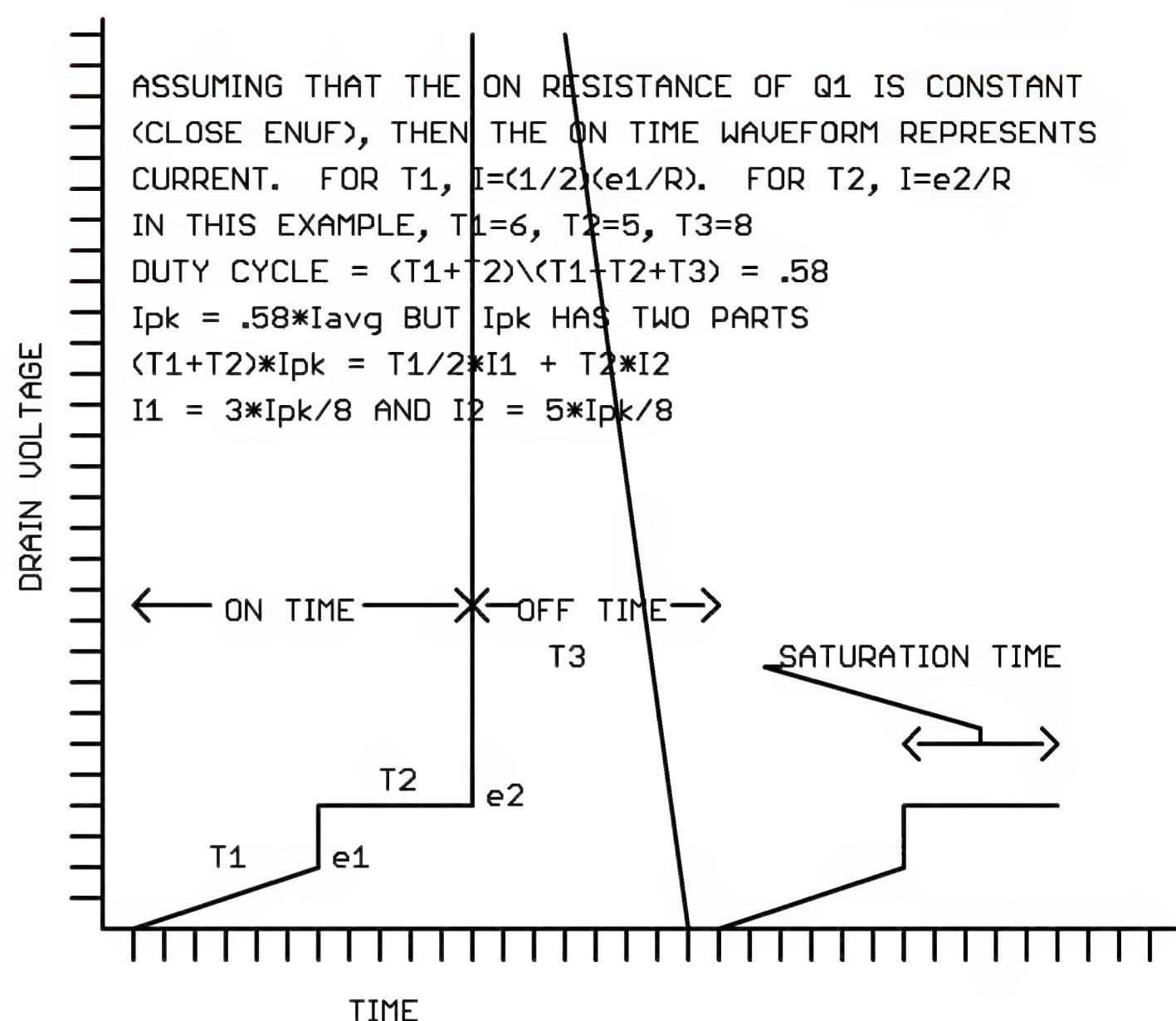
The smaller coil apparently stores less energy but the field collapses faster, so it produces a higher voltage but less current.

If you use a 555 one shot for your switch circuit (you will find the circuit and instructions on its datasheet) and power the U1 circuit from the 555 output, the power increase will be minimal.

I doubt that your circuit is stressing the 12V 5A supply, but more voltage input will give more voltage output. However, you run the risk of internal breakdown of the insulation of the coil if you exceed 14 volts at the coil. I assume that you are using an oscilloscope to measure the voltage loss at the coil because an analog meter will only measure the average, which depends on the duty cycle; a digital meter won't give any useful reading.

The waveform at the drain of Q1 should be a ramp as the current in the coil increases with a sudden rise

■ **FIGURE 2.**



to two volts as the coil saturates. You don't want the coil to remain in saturation because it just wastes power and heats Q1. Reduce the pulse width to minimize the time in saturation. You can measure the duty cycle and the average current from the source and calculate the peak

current; see **Figure 2**.

For excellent reliability, the voltage rating of electrolytic capacitors should be double the applied voltage, but for hobby use you can relax that because the caps normally exceed their rating and a failure is merely a nuisance.

Video Monitor Alarm

Q I have been reading *Nuts & Volts* for over 13 years now and have greatly enjoyed the Q&A section. I now must pose a question that I'm hoping you can answer.

I'm using a 1.2 GHz 30 IR LED wireless camera and receiver to monitor my backyard both day and night. The receiver has two RCA output jacks: audio and video. I am looking for a circuit that can generate a trigger from the video output when something new comes into the field of view. The trigger would either be used to activate a recording device or visual signal such as an LED. I have an MC9S12 Dragon12 development board and have been trying to use

the analog-to-digital converter input to monitor the analog video output from the receiver.

— **John Foster**

A I am sure that it is possible to do what you are attempting using digital signal processing, but I'm an analog engineer and have no idea how to do it. What I would do is rectify and filter the analog video signal and use a window comparator to set the alarm if the level went higher or lower.

The time constant of the filter would need to be short enough that the change from daylight to dark would not cause an alarm, and some way to deal with passing clouds would be needed. The DSP (digital signal processor) would need to deal

with those issues also.

A better solution in my mind is to use a motion detector — preferably one based on heat so that moving branches due to wind will not be detected.

Can't figure out that pesky circuit or don't understand the components? Let Russ help! Send any questions and/or comments to: Q&A@nutsvolts.com

Best Robotic Communication

Q I am building a robot using a model 2 Raspberry Pi as its brain and would like to add two transceivers with a 1,000 yard range for monitoring and controlling the robot. I would like to know the best way to do this.

I would like the first transceiver to have two USB ports for the keyboard and mouse; then, be able to receive the HDMI signal from the second transceiver and display it on a HDMI monitor. The second transceiver should be able to receive

the signal from the keyboard and mouse, sending the information by way of the USB to the Raspberry Pi.

— Timothy Harner

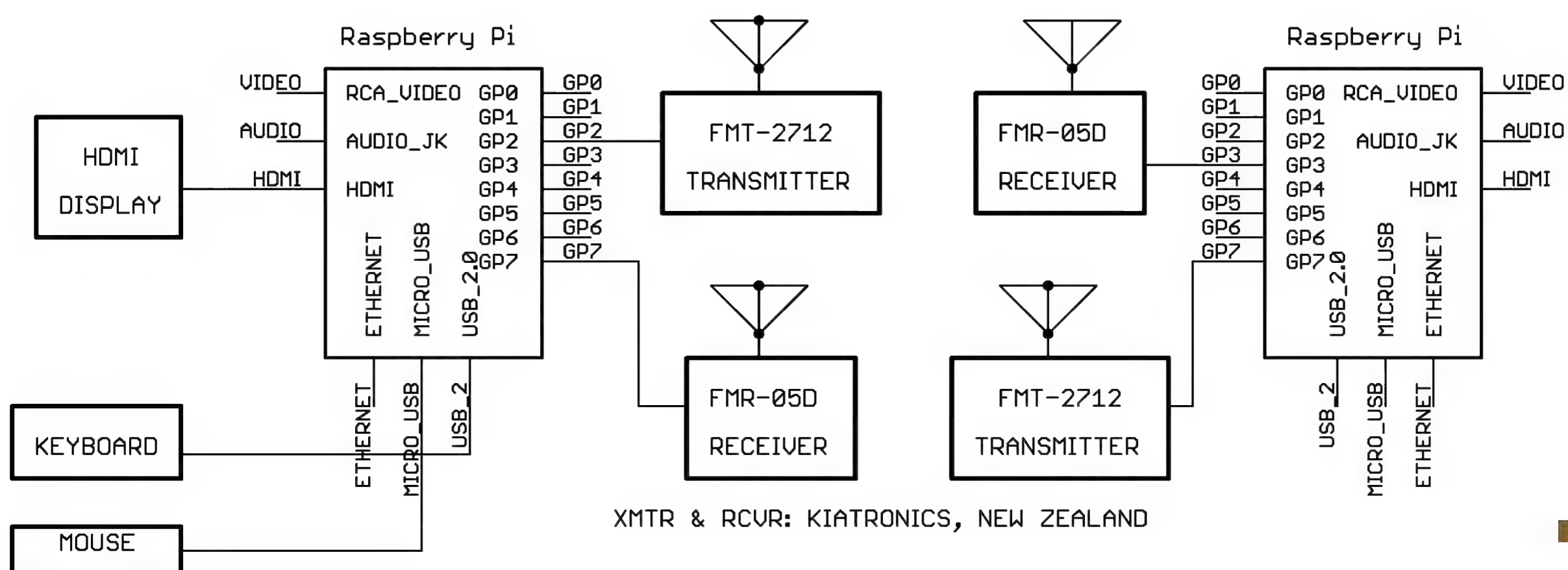
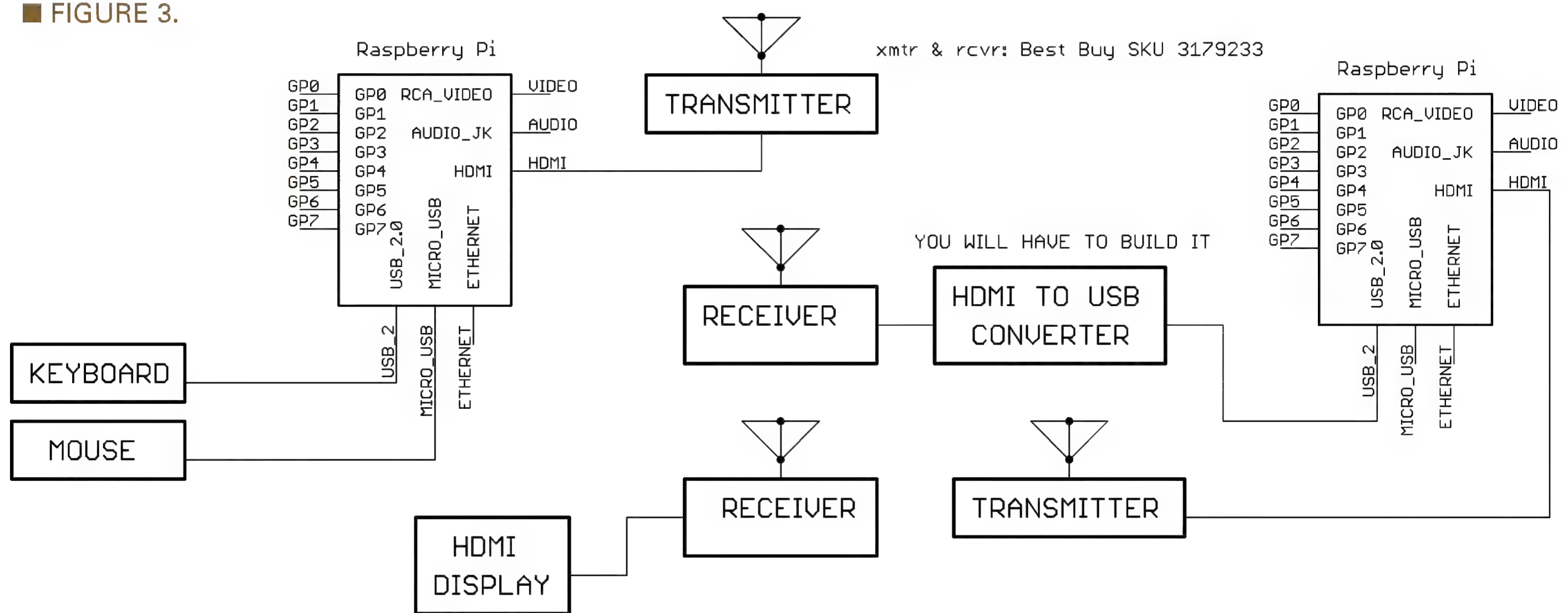
A There are many USB and HDMI capable transmitters and receivers available on the World Wide Web, but their range is limited to about 300 feet due to the FCC's (Federal Communications Commission) limitation on power output. The Raspberry Pi has two USB ports and an HDMI output plus a GPIO port, so you could have a configuration like **Figure 3** with a 300 foot range.

The only way to have more range

that I am aware of is to use the 27 MHz citizen's band which has six data channels and allows a maximum of five watts. There is a one watt module made in New Zealand which claims a range of 3,000 meters. You will have to build the interface yourself; I suggest Manchester coding because you can do that in software and use the GPIO port of the Raspberry Pi (see **Figure 4**).

If you want full duplex communication, you will need to put the two transmitters on different channels. Otherwise, you could turn off one transmitter while the other is transmitting for simplex communication. **NV**

■ FIGURE 3.



■ FIGURE 4.

MAILBAG

Re: Cat Alarm, pages 23 and 24, August 2013:

Thanks for the previous help! I've got a few more questions if you don't mind answering them:

1) I think I understand everything in the schematic except for the RF transmitter pins. Why is the out of the LMC555 connected to the ground of the transmitter? Also, what is the input pin used for? I tried looking at the datasheet but it didn't explain too much.

2) I have started to try and tackle the receiver schematic, as well (Figure 4A in the August column). I looked up the 4013N chip and discovered that it is a flip-flop, and read about what each of the pins do. I still do not understand, however, why the two Q\ output pins are connected in Figure 4A.

3) Also, why are the clock and data inputs connected to ground? I know the clock input is rising edge triggered, but is this merely an alternative form of triggering the trip? For the data input, I am entirely confused what this does.

4) Why are there two resistors (R1 and R2) between the RF receivers out and the 4013N's pins? Couldn't this be accomplished with just one?

Thanks again and if you know of any good resources on these topics for beginners, I would really appreciate it
Bill Blackburn

1) The output of the 555 goes high and low because it is connected as an oscillator. When the output is low, it is like a switch to ground which applies power to the transmitter. The transmitter input is connected to VCC, so it will produce maximum output when it is active.

2) The Q and Q/ outputs are connected in order to provide more drive to the transistor Q1. It is possible to do this when the two devices are on the same silicon chip, so they are identical. If you were to try this on separately packaged devices, the switching times would be different and the power dissipation would be increased.

3) The clock and data inputs are not used, so they are connected to ground. It is not wise to leave any input floating because stray electric fields can change the logic level and give unexpected results. I am using the 4013 as a set-reset flip-flop. The output of the receiver sets the Q output high and the reset switch sets the Q/ high. Whatever logic level is on the data input when the clock goes high is transferred to the Q output. The only time I use the data input is when I want a toggle. If you connect the Q/ to D, then every time it is clocked it changes state.

4) You are right; R2 should have connected to the reset line because it is otherwise floating. Mea culpa.

I have not looked into resources for beginners but I am sure readers will have some suggestions; I will let you know.

Re: ESD Instrument Circuits, page 16, September 2013:

Mr. Kincaid asked about measuring static electricity: "Two foils will separate when charged due to electric

repulsion, but I don't know how to implement that. Does anyone have any ideas?"

I do! Ahem ... I might! What about reflecting a beam of light off one of the leaves which are shiny metal to start with; if I recall properly, displacement of the leaf would create a proportional displacement of the beam. Said beam is hitting — depending on the position of the leaf — somewhere between the edge and the center of a loupe lens. The lens is located over a photoresistor. The light beam ending up striking the photoresistor varies in its area depending on its position, varying the actual resistance of the photoresistor. Appropriate amplification shall be implemented downstream, of course.

Would that work? Run with it if you want; I am just glad I could finally think of something — anything — for a *Nuts & Volts* question. LOL!

I'm always glad to read your columns. I signed up this year, thanks to having acquired an iPad which lets me read *N&V* at my leisure.

Stephen Young

Yes, that will work. It will be nonlinear and subject to vibration, but lacking any other suggestion it is a good one.



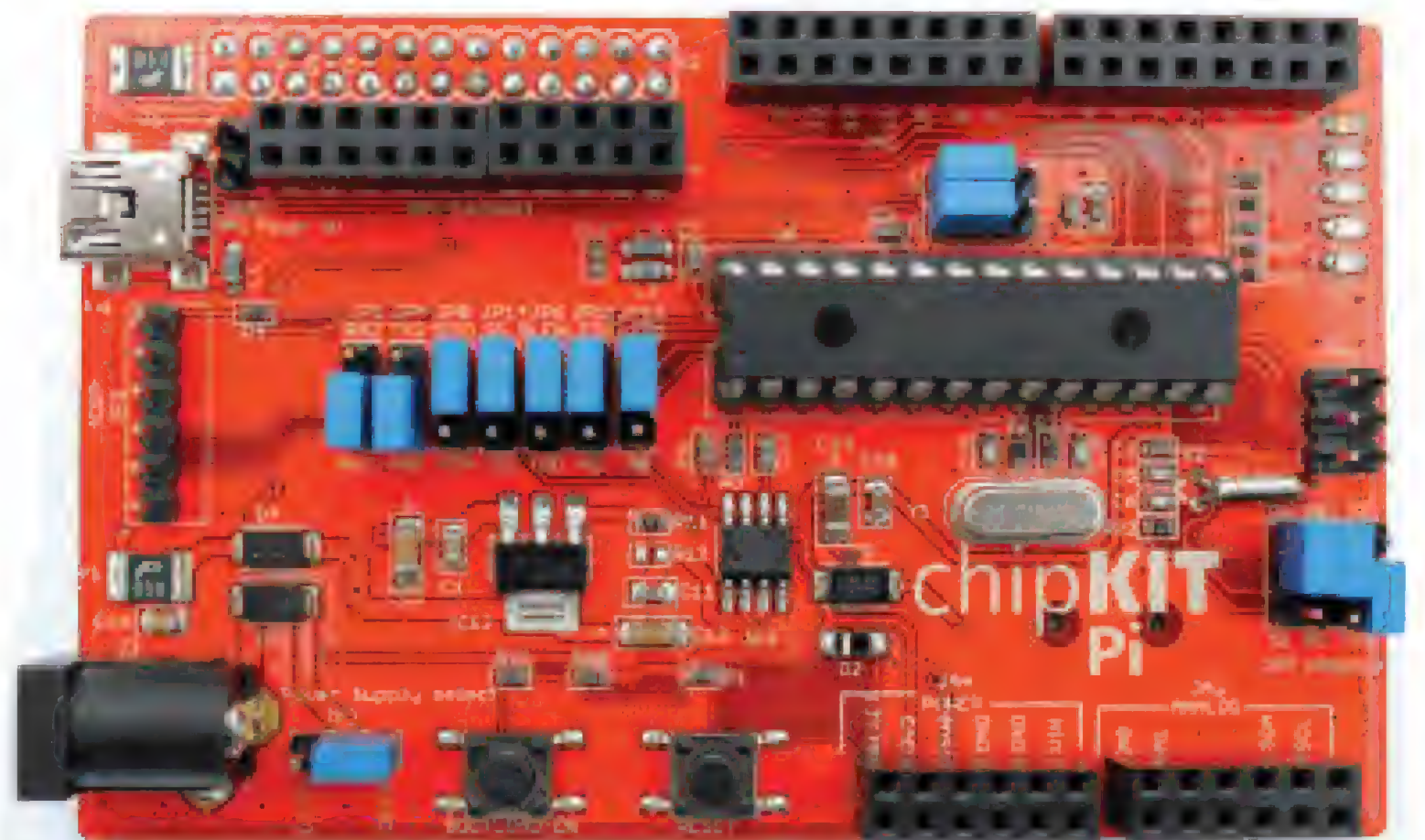
NEW PRODUCTS

- HARDWARE
- SOFTWARE
- GADGETS
- TOOLS

RASPBERRY PI/chipKIT EXPANSION BOARD

Microchip Technology, Inc., announces the expansion of its Arduino compatible chipKIT platform ecosystem, including a new Raspberry Pi tool that it co-developed with partner element14 — the chipKIT Pi Expansion Board. On the software side, volunteers from the chipKIT and Arduino communities collaborated with Microchip's engineers to expand the free chipKIT Multi-Platform IDE (MPIDE) to allow users to create, compile, and program Arduino sketch-based chipKIT applications within the Raspberry Pi operating system.

The chipKIT MPIDE is open source and compatible with the Arduino programming language and development



environment. Both of these tools are based on Microchip's 32-bit PIC32 microcontrollers in prototyping-friendly, low

Superior Quality Soldering Tools from

HAKKO

NEW



FX-951

HIGH PERFORMANCE SOLDERING STATION

- Digital temperature display
- Superior heat transfer
- Fast thermal recovery
- 400°- 840°F (200°- 450°C)
- Low temperature alarm
- Sleep mode for longer tip life



FX-601

TEMP-ADJUSTABLE SOLDERING IRON

- Electronic "closed loop" control
- 464°-1004°F (240°- 540°C)
- Ceramic heating element
- Lightweight (2.4oz w/o cord)
- Extra long 3-plug, 6 ft.cord
- UL/CSA Approved



FX-888D

LOW COST, DIGITAL SOLDERING STATION

- Digital display (°F or °C)
- Adjustable temperature
- 120°-899°F (50°- 480°C)
- Idle temp within 1.8°F (1°C)
- Ergonomic, handpiece
- Ceramic heater



For complete information, visit HakkoUSA.com/NutsVolts

pin count SPDIP packages, which was previously only available with eight-bit MCUs for the Arduino community. This enables all users — including hobbyists, academics, makers, and professionals — to benefit from the PIC32's high performance, memory, and integrated peripherals while using the basic hobbyist prototyping equipment that is found in most home workshops.

Because the Raspberry Pi's processor is a 3.3V chip, its digital I/O and communication (I²C, UART, SPI) pins require 3.3V. Meaning that existing Arduino boards — which are all based on 5V eight-bit MCUs — require users to create or purchase additional components to translate the voltages. The chipKIT Pi can interface directly to the Raspberry Pi I/O expansion header without any additional components, reducing both cost and design complexity.

Previously, hobbyists, academics, and professionals who wanted to develop Arduino applications for the Raspberry Pi were forced to use boards based on eight-bit MCUs. The chipKIT Pi's 32-bit PIC32 is in an SPDIP package, so users can swap out the pre-populated PIC32 with a new one without having to replace the entire board.

The PIC32's high level of performance, memory, and integrated peripherals allow users to create applications with greater functionality, including touch sensing, audio processing, and advanced control. Additionally, users can take advantage of the PIC32's peripheral pin select feature which provides the flexibility to re-map certain peripheral I/O pins to suit their applications.

The chipKIT Pi board and the Raspberry Pi compatible version of the MPIDE software further enable users to tap into the large repository of available Arduino tutorials, reference materials, curriculum, and more to create a diverse array of designs.

element14's chipKIT Pi Expansion Board is priced at \$28. Additionally,

the Raspberry Pi-compatible version of the chipKIT MPIDE can be downloaded for free at www.microchip.com/get/EU61.

For more information, contact:
Microchip Technology
Web: www.microchip.com

PARABOLIC MICROPHONE FOR AUDIO SURVEILLANCE

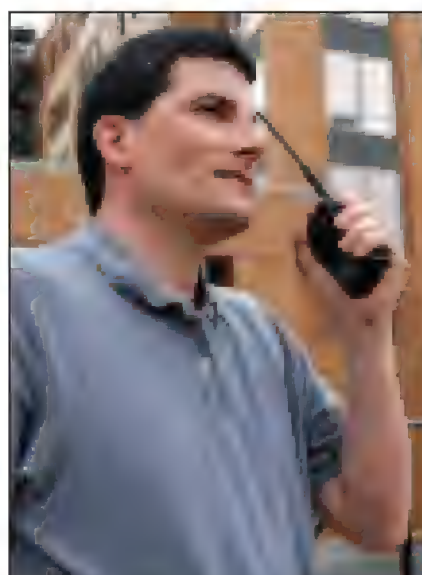
MJ Electronics is now offering an updated version of their advanced electronic parabolic microphone — the P650.

Developed over a three year period and constantly updated, this broadcast media quality parabolic microphone is in use around the

world by federal and local law enforcement, surveillance professionals, professional nature observers, and documentary film producers.

This long range parabolic microphone is capable of picking up and magnifying audio

signals up to 75 times that of a normal omni-directional microphone while remaining virtually impervious to overload. Powered by two nine volt batteries, the P650 assembles ready for use in less than two minutes and comes complete with a built-in microphone capsule, equalizer/amplifier in the handle, earphone, aiming sights, effects/voice switch, and a one year manufacturer's

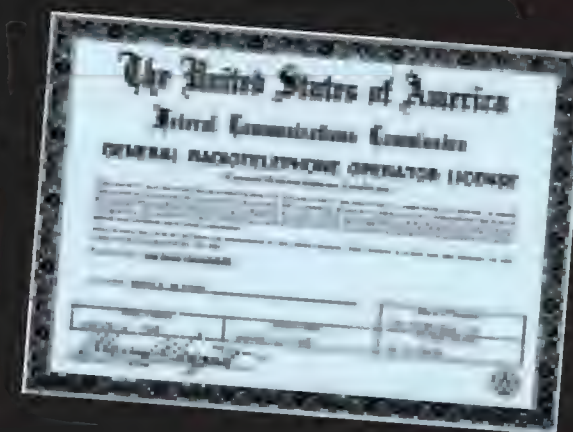


EARN MORE MONEY Get your dream job!

**Make up to \$100 an hour
and more as an FCC
Licensed Technician!**

**Get your "FCC Commercial Radio License" with
our proven Home-Study Course!**

- No costly school. No classes to attend.
- Learn at home. Low cost!
- No experience needed!
- **MONEY BACK GUARANTEE:** You get your FCC License or money refunded!



Turn your interest in electronics into a high paying career in Communications, Radio-TV, Avionics, Marine, Radar, Safety, Municipal and more... even start your own business!

Call for FREE info: 800-932-4268

Or, email us: info@LicenseTraining.com

ext. 109

www.LicenseTraining.com

COMMAND PRODUCTIONS FCC License Training

Industrial Center, 480 Gate Five Rd., PO Box 3000, Sausalito, CA 94966-3000



warranty. Price is \$610.

HD 1080 SPORTS CAMERA CAM SJ APPLICATIONS

Also from MJ Electronics is their new waterproof action camera.

The SJ high-definition sports digital video camera is ultra-slim and has an integrated look design. Its motion capture technology is able to clearly record all sorts of activities.



This product can be used for a wide range of purposes and is an essential recording medium for vehicle data recording, outdoor sports, home security, and deep-water probing applications.

Main features include:

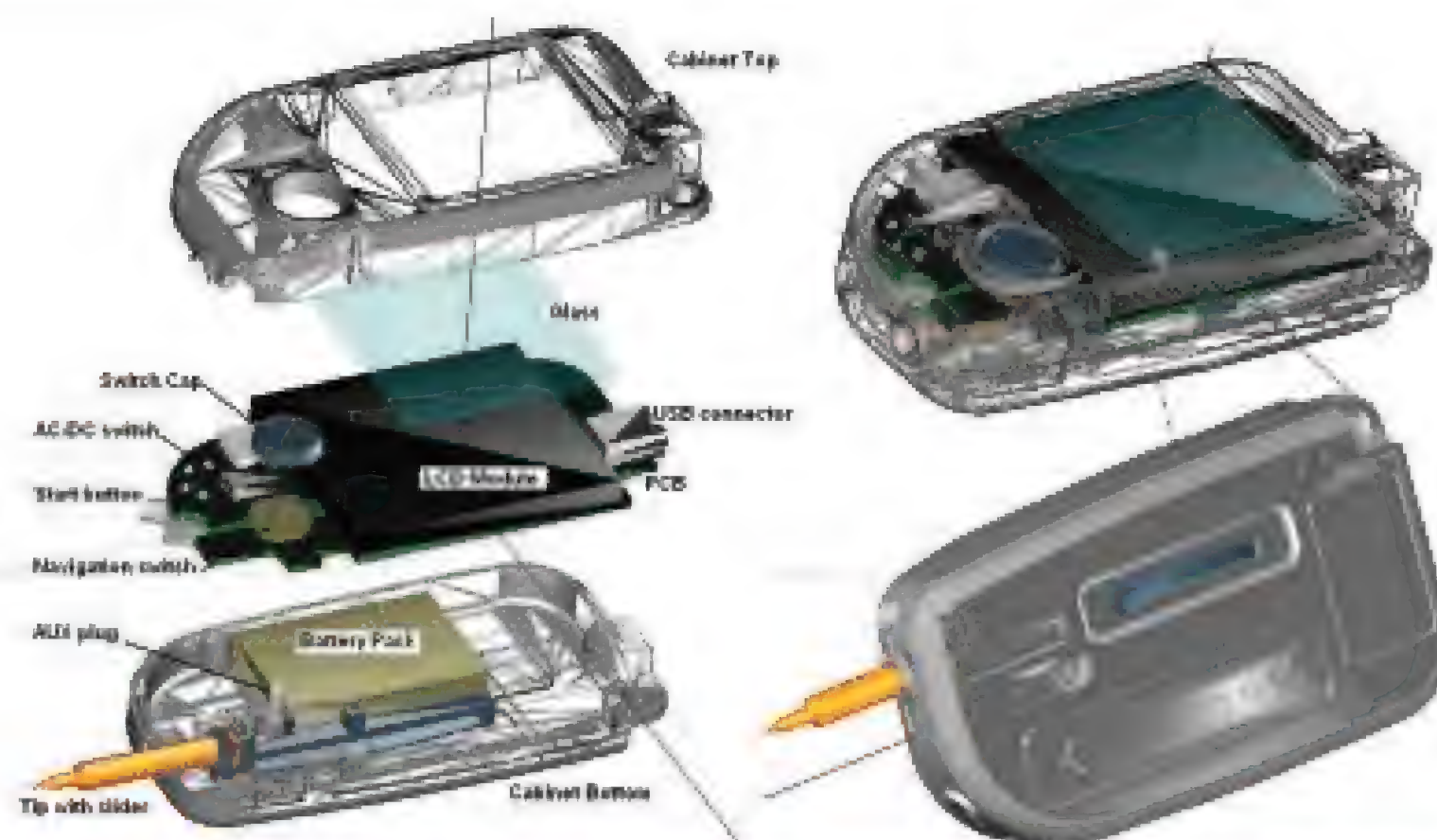
- Video capture feature.
- A 1.5 inch high-definition screen that displays and replays recorded videos.
- Detachable battery that is easy to exchange.
- Video recording while charging.
- Plug-in video recording.
- Twelve megapixel HD wide-angle lens.
- HDMI HD output feature.
- Web camera.
- Maximum 32 G storage card (does not include a memory card in parcel).

Price is \$115.

For more information, contact:
MJ Electronics
Web: www.mjelectronics.com

DIGITAL STORAGE OSCILLOSCOPE

Key-Scope – potentially the world's smallest digital storage oscilloscope with color LCD – has been introduced through KickStarter



and is available for pre-sale through Wittig Test Technology, Inc.

The new digital storage oscilloscope for mobile measurement tasks and easy PC connection is small, but performs as a full functioning oscilloscope. Priced below US\$60, it serves as a useful gadget or gift to the electronics hobbyist. Since all major functions of a traditional oscilloscope are covered with Key-Scope, it also serves as a useful tool for electronics students to get familiar with oscilloscopes. Key-Scope is redesigned and based on the formerly released Wittig osziFOX.

Probe Scope is another product that is pre-selling which offers trigger auto, pre/post trigger positioning, trigger level adjustment, and internally/externally and negative/positive trigger transition.

The measured signals are refreshed on the unit's small color LCD and the PC application via USB simultaneously and continuously.

The package includes an auxiliary and ground cable with clip, a USB cable, and a manual.

For more information, contact:
Wittig Test Technology
Web: www.Wittig.it or
www.Kickstarter.com

PIC BASIC PROJECT PCB

Images Scientific Instruments is now offering a PIC project PCB which features an LCD display with backlight and contrast control; two

Recycling & Remarketing High Technology

WEIRDSTUFF[®]

WAREHOUSE

Software, Computers, Electronics, Equipment, Doohickies

WE BUY/SELL EXCESS & OBSOLETE INVENTORIES

FREE COMPUTER AND ELECTRONIC RECYCLING

GIANT 10,000 SQ. FT. AS-IS SECTION

**384 W. Caribbean Dr.
Sunnyvale, CA 94089**
Mon-Sat: 9:30-6:00 Sun: 11:00-5:00
(408)743-5650 Store x324
WWW.WEIRDSTUFF.COM

www.Primecell.com

Battery rebuilding service

Dead Batteries? Don't toss them.
Send them to us - our rebuilds are
better than original specifications.

Tools Hitachi Milwaukee Panasonic B&D DeWalt Makita All 2-36 Volts	Electronics Bar Code Scanners Surveying Printers Laptops Photography	Radios APELCO UNIDEN G.E. ICOM KENWOOD MOTOROLA MIDLAND MAXON YAESU ALINCO
---	---	--

Visit www.primecell.com for important details
24 Hr Secure recorder tel-fax (814) 623 7000
Quotes email: info@primecell.com
Cunard Assoc. Inc. 9343 US RT 220 Bedford PA 15522

Small Mechanical Components for

ROBOTICS

800-819-8900

SDPSI Visit sdp-si.com
and Buy Online Today.

ROBOT POWER *Extreme Motor Speed Control!*
MegaMoto - Motor Control for Arduino™

- ◆ 6V-28V - 13A / 30A peak
- ◆ Single H-bridge or dual half
- ◆ Current and Temp protected
- ◆ Jumper select Enable and PWM pins
- ◆ Up to three units stack on one Uno/Duemilanove

Dalf - \$250

- ◆ Closed-loop control of two axes
- ◆ Windows GUI for all features
- ◆ Full PID position/velocity loop
- ◆ Trapezoidal path generator
- ◆ Giant Servo Mode!
- ◆ www.embeddedelectronics.net

>> Also H-bridges, gearmotors, plans, schematics <<
www.robotpower.com
Phone: 253-843-2504 • sales@robotpower.com

LE MOS

INTERNATIONAL

ZigBee Home Automation

ZigBee Pro
ZigBee Range Finder

www.lemosint.com

Quality Kits for All Your Electronic Needs!

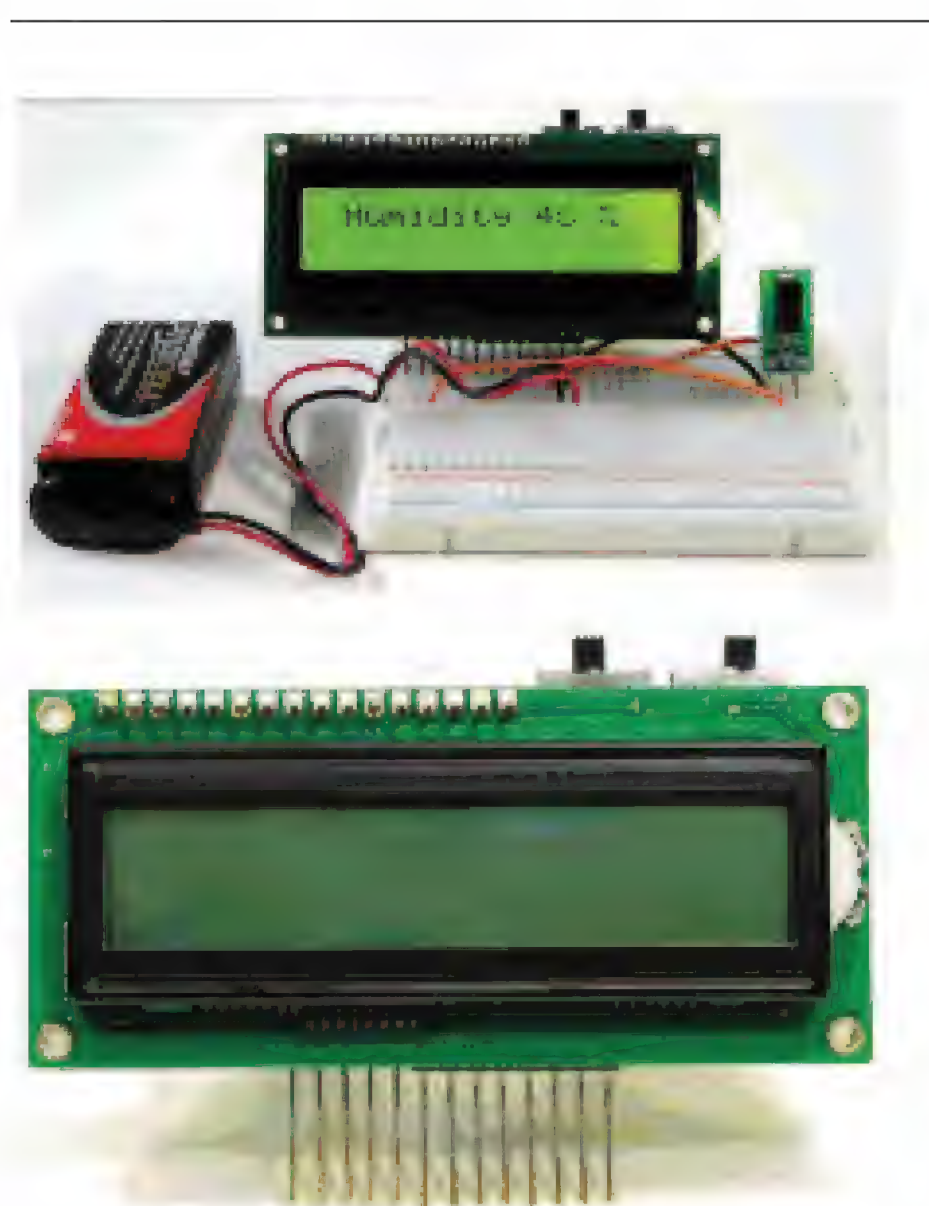
Explore the World of Electronic Project Kits and Devices.

LOTS OF GREAT GIFT IDEAS!

We offer a wide selection of electronic kits for beginners, as well as more advanced projects for students and experts!

We also carry Arduino and Raspberry Pi products, as well as a variety of speed controllers, timers, power supplies, plus lots more!

QKITS.COM



A/D channels; eight digital I/O lines; 5V and 9V operation; solderless breadboard; and a free student version of PICBASIC PRO and Microcode Studio. Price is \$19.88.

For more information, contact:
Images SI, Inc.
Web: www.imagesco.com

In the Nuts & Volts Webstore NOW!

NUTS AND VOLTS

NV BOOK SPECIALS

HACKING ELECTRONICS

309 Pages
Only
\$23.95

Raspberry Pi
Projects for the
EVIL GENIUS

224 Pages
Only
\$19.95

SAVE 20%!

To order call 800 783-4624 or visit:
<http://store.nutsvolts.com>

Limited time offer

SLIDE MOUNT for MOBILE RADIOS

UNIVERSAL: FITS MOST 2 WAYS

Model ET-850 Universal Slide Mount. Standard mounting platform, specify radio make/model. Ends blind fumbling to release hidden fasteners. Easily transfer radios from vehicle to vehicle. Also use in the office, job trailer, marine too. Black powder coat finish. Volume prices.

(Ask about our maximum security Locking Radio Cage™ Mount)

E TIP, Inc., North Aurora, IL 60542 USA
(800) 530-5064 FAX (630) 801-9569
www.etipinc.com | sales@etipinc.com

E TIP is a Veteran Owned Small Business

NKC electronics

MDE8051 Trainer by Digilent

NKCElectronics.com/MDE8051

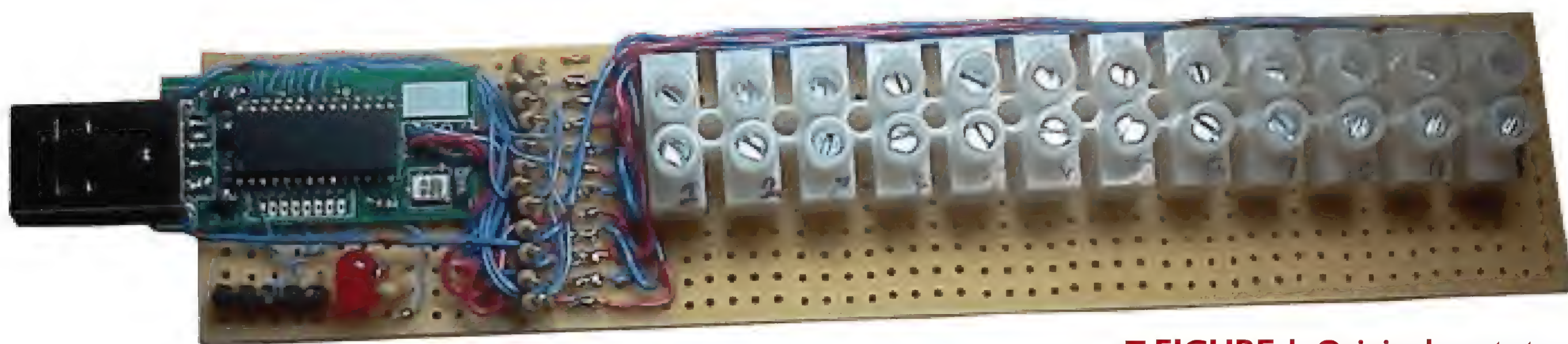
Includes the MDE8051 training board, power supply, serial cable

Purchase Orders are accepted from Educational Institutions, US Government and Research Centers

QUICK AND EASY USB KEYBOARD INPUT

By William Pippin

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?/magazine/article/november2013_Pippin.



■ **FIGURE 1.** Original prototype.

Ever want a physical trigger to manipulate keyboard inputs? Then, this is the circuit for you.

A while ago, I was asked by a friend if I could design a simple USB input device to allow substituting keyboard inputs with mechanical micro-switch inputs to operate his TIC-TAC-TOE video game. It was written in BASIC and he didn't want to re-write any of the program. He also had a wooden box with nine [in a 3x3 matrix] square holes through which a player would throw a beanbag. As the beanbag went through the opening, a small switch in the bottom of the hole would trigger a contact closure.

He then wanted that to somehow trigger the keyboard keys from 0-9, and a foot pedal to send a <backspace> character to reset the game. He had taken an old keyboard and wired into that, but it was clunky, broke easily, and didn't look very nice when he took the game to retirement centers to entertain the people there.

Throw a beanbag through a wooden slot and get a keystroke on a PC. Yes, engineering usually comes in the form of simple problem-solving. So, here goes — let's solve this problem.

Implementation

Now that we know what this project is meant to do, let's get into the technical aspects of implementation.

I've been designing circuits for a long time now, so I have a sizeable stash of circuit building blocks which get used over and over because they are well known to me and work every time. I'll be using one of these general-purpose PCBs (printed circuit boards) that's based on a Microchip PIC18LF2550 that pretty much acts like a small USB thumb drive with several pins of analog or digital I/O. **Figure 1** shows my off-the-shelf PCB being used in the original prototype.

Notice the simplicity and flow of this design. Starting on the left, we have our connection to a PC or USB cable, a five-pin programming port,

an LED to indicate basic functionality, and a screw terminal array configured to behave as digitally compatible switching inputs via some pull-up resistors. I don't mean to understate things, yet the circuit design is just that simple. This is certainly a case where the firmware handles the brunt of the task.

All project files are available at the article link, so the schematic diagram presented here is to be used as an outline of behavior. Notice how the schematic generally resembles the image of the original prototype.

There are plenty of standards and opinions regarding circuit design. Keeping things simple and straightforward is usually the best policy to promote understanding and even dreaded troubleshooting. The schematic's only difference is it has more input switches in the form of pull-up resistor-connected pin headers (top of schematic: S1-S16) and the addition of an optional voltage regulator (IC3).

The switches work as simple closure contacts. For example, R2 (1K ohm to 39K ohms, depending on signal skew and transient rejection desired) is connected serially to the voltage source and to switch S1 (pin header). The other connection is S1 to ground.

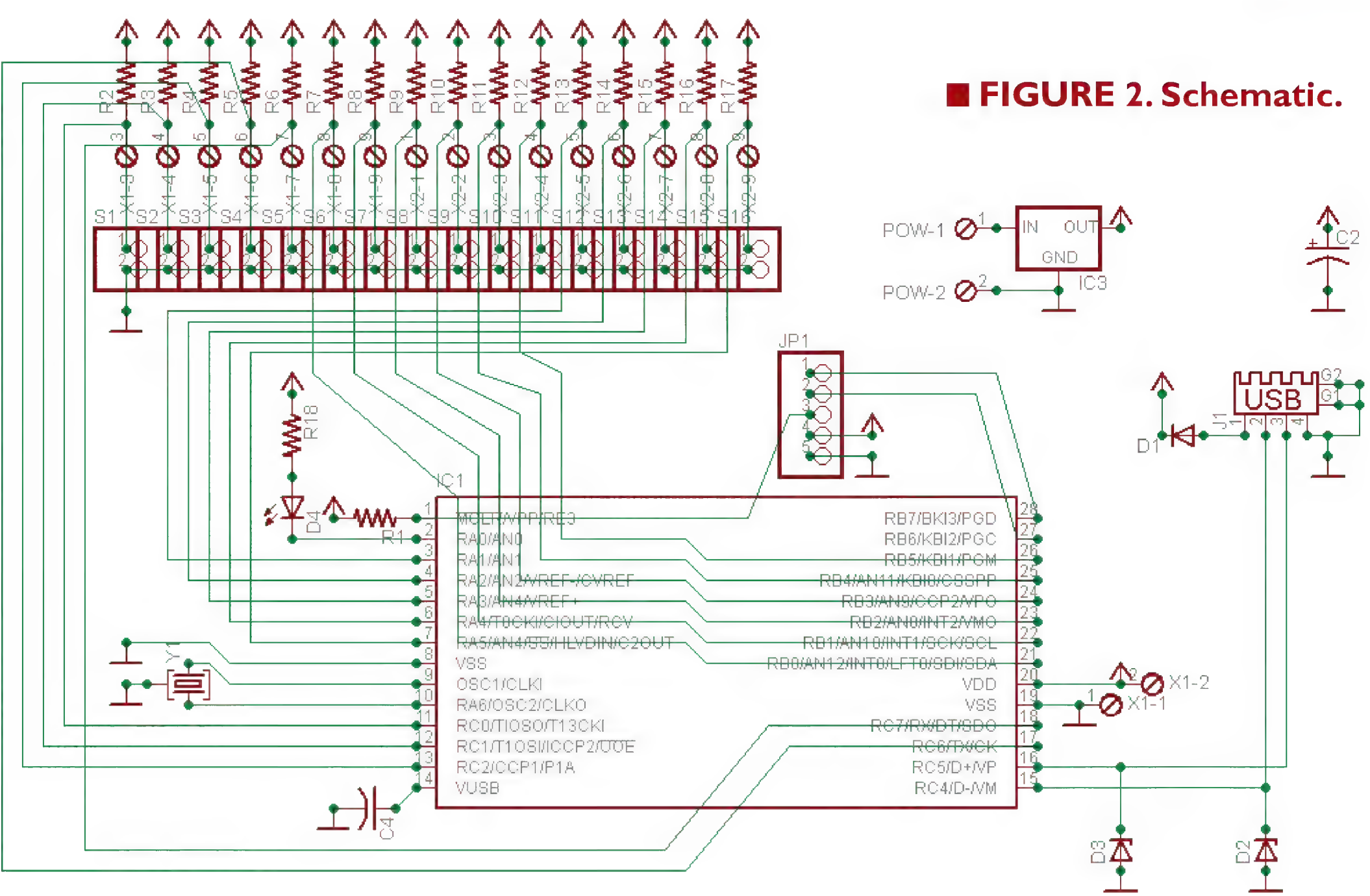
When S1 is closed (terminal is shorted to ground), the voltage seen at the junction of R2 and S1 changes from

the nominal high voltage (five volts, in this case) to ground (0 Vdc). The PIC will interpret this voltage change as an input logic transition from 1 to 0. Thus, we have a negative logic switch. As long as our firmware knows this, everything will work fine.

Resistor R18 (1K ohms-5K ohms, depending on desired brightness) and diode D4 are the components used to connect an "idiot light" LED to our PIC.

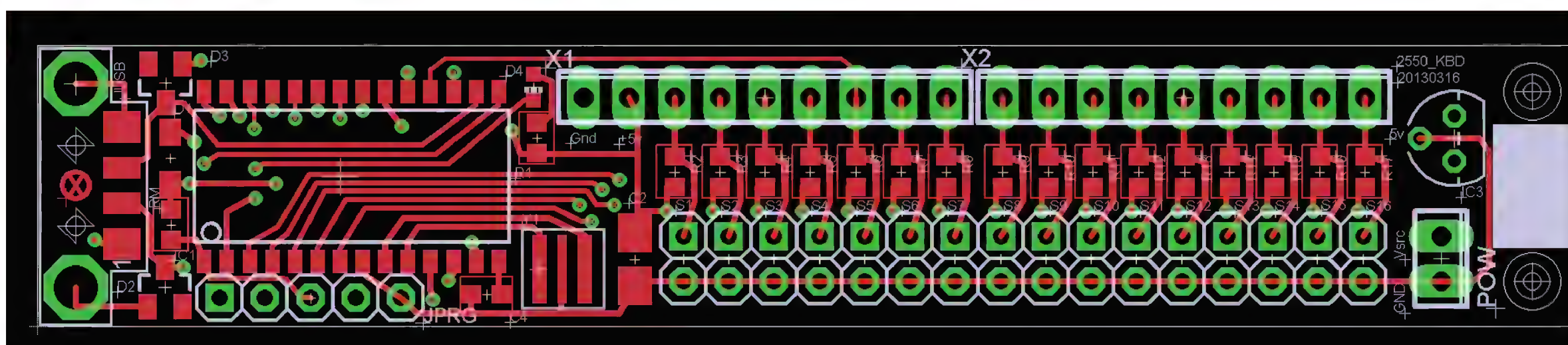
Again, this will be a logic 0 driven output. Other than the output via a USB connection, this will be our only indication of functionality.

The example firmware is written so that the LED will illuminate on device startup initialization and on key presses (switch closures). In my humble opinion, there



ITEM	DESCRIPTION	MFG PART#	SOURCE / NOTES
C2	CAP/TANT 4.7 μ F 16V 20% 1206	F931C475MAA	Nichicon
C4	CAP/CER 0.47 μ F 10V 20% 0603	C1608X5R1A474M	TDK Corporation
D1	DIODE SWITCHING 75V SOD323	1N4448WS-7-F	Diodes, Inc.
D2-3	DIODE ZENER 3.9V SOT23-3	BZX84C3V9-7-F	Diodes, Inc.
D4	LED CHIPLED 633NM RED 0603	LS Q976-NR-1	OSRAM Opto Semiconductors Inc / Not required for operation
IC1	IC MCU 8-BIT 32 KB FLASH 28SOIC	PIC18F2550-I/SO	Microchip Technology
IC3	REG LDO 5V 0.1A TO92-3	L78L05ACZ	ST Microelectronics / Not required for operation
JUSB	CONN USB A SMD	1001-011-01101	CNC Tech
JPRG	CONN HEADER .100 5-POS	PEC05SAAN	Sullins Connector Solutions / Programming Header
POW	9V BATTERY CONNECTOR	84-4	Not required for operation
R1-17	RES 39.0K OHM 1/10W 1% 0603	RC0603FR-0739KL	Keystone Electronics / Not required for operation
R18	RES 3.30K OHM 1/10W 1% 0603	RC0603FR-073K3L	Yageo / Valid range from 1K ohm to 100K ohm
S1-16	MOMENTARY SWITCH SPST-NO	GPB024A05BB	Yageo / Valid range from 100 ohm to 5K ohm
Y1	20 MHz CERAMIC RESONATOR	PBRC20.00MR50X000	Not required for operation

Note: Generally, most of these part values and manufacturer part numbers can be substituted. This circuit is very forgiving! So, use whatever you have around as long as it is reasonable.



■ **FIGURE 3.**
PCB layout.

isn't much else that would be useful in terms of basic troubleshooting.

The rest of the components are supporting hardware in order to make the PIC happy. Y1 is our ceramic resonator clock source. JP1 is a pin header used for in-circuit programming. Resistor R1 (33K ohms-39K ohms) is the MCLR (programming voltage) pull-up resistor. C2 is a decoupling capacitor with a value of 1 μ F to 5 μ F. C4 is a ceramic capacitor used by the PIC18LF2550's internal regulator. There is a range of values that C4 can have, but I've found 0.47 μ F works the best in most situations.

D3 and D4 are 3.9 Vdc zener diodes used as transient and over-voltage protection for the USB data lines. There are many ways to perform this task. I had a bunch of zener diodes, so I used them. D1 is a 1N4148 type used to isolate the PCB voltage from the USB attached to the PC. If we are using the power from the USB connection, leave D1 connected as shown.

If, for some reason, you choose to use a separate power supply in addition to the PC, remove D1 from the circuit. This will disconnect the PC power from the circuit and allow our own board voltage sources of IC3 or

(directly) X1 to power our device. The design used here has the USB as the power source, so as not to damage the PCB or any attached USB host when using optional power connections.

Figure 3 shows our EAGLE produced PCB layout. Notice there is a row of pin headers on the top and a double row on the bottom. The top row is connected only to the junction of the switch pull-up resistor and the switch. These are used in the case where your switches use a shared ground. The bottom double row is used for two wire connected momentary switches as shown in **Figure 4**. Additional power options are on the right of the PCB.

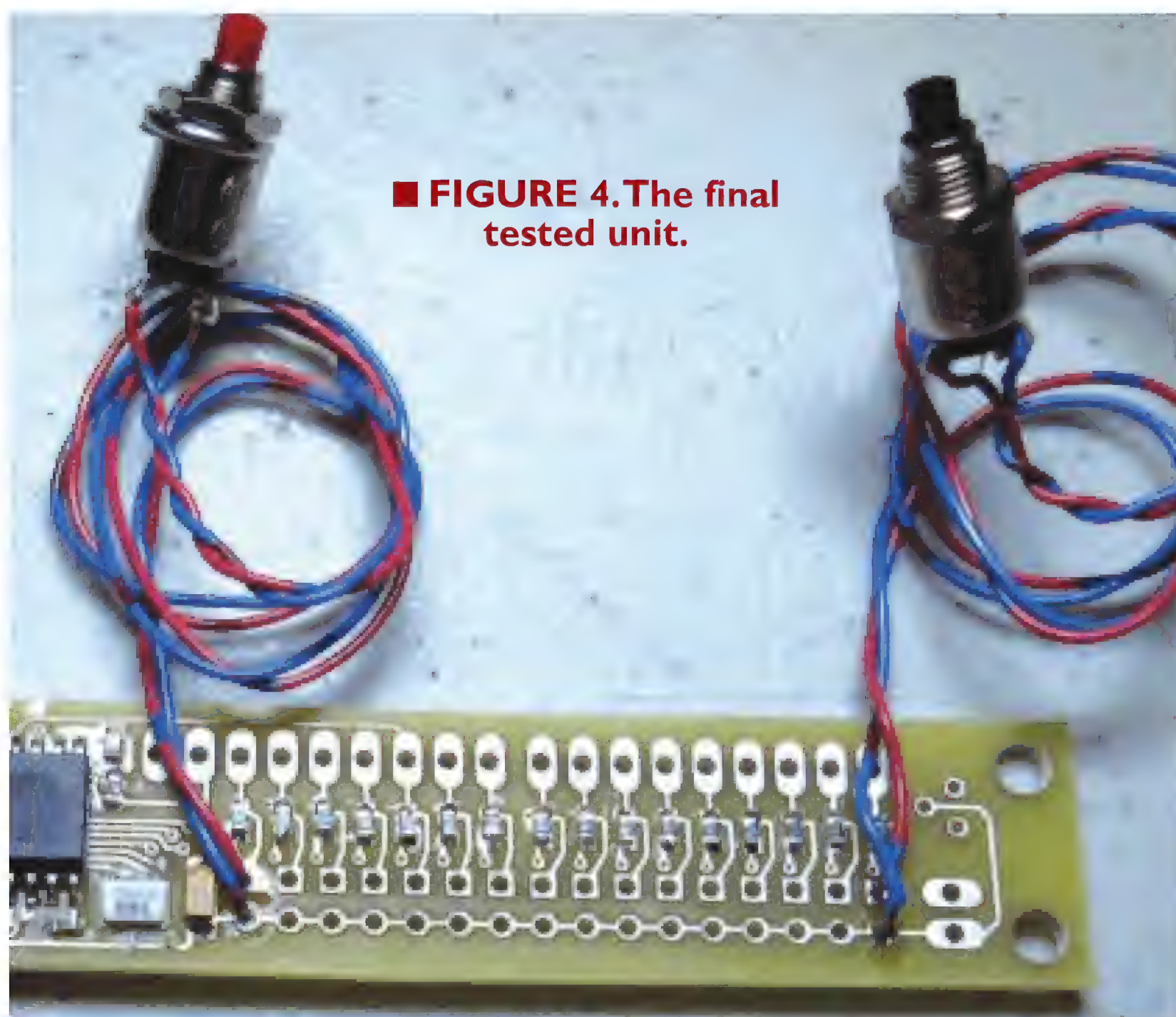
Firmware

The final piece of the puzzle is that the PIC/PCB needs to act like a HID keyboard device. Luckily, we don't have to start from scratch because my PIC compiler of choice — CCS' PIC-C — has been good enough to provide some example USB HID files that we can quickly modify in order to get our device up and running. These files begin with "usb_kbmouseXXX.c" and are within the PCW USB directory of the compiler.

If you do not have this compiler, no big deal. The concept is always the same: Tell the attached PC what kind of HID device we are via the device descriptor, and send messages to (in other cases, receive from) the PC that are accepted as HID type messages. Beyond that, we probably want to tell our PIC what messages (key/scan codes) to send, depending on which key closures are shorted to ground. The provided firmware sends these key/scan codes: 0-9, a-d, <return>, and <backspace>.

Figure 5 is a screenshot of the main function of our PIC18LF2550's firmware code. We'll go over the main function first and its generalities, followed by a couple of smaller sections of code that aid in understanding what is actually happening.

Lines 264-282 get the PIC up and running with configuration and initialization code related to setting port pins as inputs or outputs, registers, and setting up the USB device descriptor.



■ **FIGURE 4.** The final
tested unit.

Line 284 calls our function *LoadEEPROMcfg()*. This function stores our scan codes into the PIC's EEPROM area. There is no need to do this. We could have just as easily hard-coded the scan codes, but this would require a complete re-compile of the source code every time we wanted to make a change.

Storing these values in EEPROM allows us to use an EEPROM editor to set the appropriate memory locations with the scan codes that we want reported on a key hit. It simply makes life a little easier in the cases where we either don't have the correct compiler or like our ROM code just fine.

Line 284 calls our function *Fill_USB_SCANCODES()*. This function validates our scan code values just to make sure that the data isn't corrupted or that our values are not out of reasonable range. Knowing my customer, I wanted to make sure that if he accidentally changed the EEPROM stored scan codes to something that was 'useless' for his needs, that the firmware could catch it and adjust as needed.

Lines 287-290 are just a short delay that lets our indicator LED stay on for a bit so we can see it. The loop also calls the internal USB service routine periodically so that we don't miss any of the USB communication events.

Lines 292-303 are the main loop infinite service routine. It is, again, pretty straightforward. We do some

```

261 void main() {
262     int8 scnt;
263
264     setup_adc_ports ( NO_ANALOGS );
265     setup_adc( ADC_OFF );
266
267     set_tris_a( 0b11111110 );
268     #use fast_io( A )
269     set_tris_b( 0b11111111 );
270     #use fast_io( B )
271     set_tris_c( 0b11111111 );
272
273     LED1_ON();
274
275     SETUP_WDT( WDT_OFF );
276     SETUP_TIMER_0( TO_OFF );
277     SETUP_TIMER_1( T1_DISABLED );
278     SETUP_TIMER_2( T2_DISABLED, 255, 16 );
279     SETUP_TIMER_3( T3_DISABLED );
280
281
282     usb_init();
283
284     LoadEEPROMcfg(); // Must be called first
285     Fill_USB_SCANCODES(); // Must be called second
286
287     for( scnt = 0; scnt < 33; scnt++ ) { // startup loop
288         delay_ms( 33 );
289         usb_task();
290     } //end startup for loop
291
292     while ( TRUE ) { // main infinite while loop
293
294         MAINLOOP_WAIT();
295         usb_task();
296         LED1_OFF();
297
298         if ( usb_enumerated() ) {
299             usb_keyboard_task();
300             usb_rx_task();
301         } //end usb_enumerated() if
302
303     } //end main infinite while loop
304
305 } //end main
306 //=====
307
308

```

■ FIGURE 5. Main function of the firmware.

housekeeping to keep the USB connection alive, and if we are connected and enumerated on the USB chain, we look for our key hits and send the scan codes as required. Clean and simple.

Now, let's look at the function *usb_keyboard_task()* in a bit closer detail. After all, it does all of our task-specific work. **Figure 6** is a screenshot of the function that handles the bulk of this work. The basic flow is: a) Lines 129-144 scan our key contact inputs for what we define as hits, in this case, a logic 0; b) Lines 147-168 build and send the HID formatted message and turn on the LED for each of our pressed keys; c) Line 170 saves some state information that helps us ignore key presses; and d) Lines 173-174 add in a little bit of delay so that our keys don't send information too quickly.

There is nothing fancy in any of this, and I'm sure there are many better and different ways to accomplish the same thing. There are only two specific things worth commenting on. The first is that the HID message packet used in this instance is eight bytes long and the key code is placed into the fourth byte (*tx_msg[3]*). If you look at the function header comment, the detail of the data packet is described. Again, I didn't do anything fancy here, although we could have been more efficient.

The second is Line 160 shows an *if* statement that compares the previous key hit to the present key hit. If it is

```

123 int8 i = 0;
124 boolean bSentKey = FALSE;
125 static int8 tx_msg[8] = {2,0,0,0,0,0,0,0};
126
127 //read all 12 buttons
128 //we could read port values instead of individual reads. no rush...
129 arrBUTTON_STATES[ 0 ] = HID_KEYPRESS( BUTTON1 );
130 arrBUTTON_STATES[ 1 ] = HID_KEYPRESS( BUTTON2 );
131 arrBUTTON_STATES[ 2 ] = HID_KEYPRESS( BUTTON3 );
132 arrBUTTON_STATES[ 3 ] = HID_KEYPRESS( BUTTON4 );
133 arrBUTTON_STATES[ 4 ] = HID_KEYPRESS( BUTTON5 );
134 arrBUTTON_STATES[ 5 ] = HID_KEYPRESS( BUTTON6 );
135 arrBUTTON_STATES[ 6 ] = HID_KEYPRESS( BUTTON7 );
136 arrBUTTON_STATES[ 7 ] = HID_KEYPRESS( BUTTON8 );
137 arrBUTTON_STATES[ 8 ] = HID_KEYPRESS( BUTTON9 );
138 arrBUTTON_STATES[ 9 ] = HID_KEYPRESS( BUTTON10 );
139 arrBUTTON_STATES[ 10 ] = HID_KEYPRESS( BUTTON11 );
140 arrBUTTON_STATES[ 11 ] = HID_KEYPRESS( BUTTON12 );
141 arrBUTTON_STATES[ 12 ] = HID_KEYPRESS( BUTTON13 );
142 arrBUTTON_STATES[ 13 ] = HID_KEYPRESS( BUTTON14 );
143 arrBUTTON_STATES[ 14 ] = HID_KEYPRESS( BUTTON15 );
144 arrBUTTON_STATES[ 15 ] = HID_KEYPRESS( BUTTON16 );
145
146 //send the values
147 for( i = 0; i < MAX_CNT_BUTTONS; i++ ) {
148
149     if ( arrBUTTON_USB_SCANCODES[ i ] == 0x00 ) //skip an input because it is disabled
150         continue;
151
152     if ( arrBUTTON_STATES[ i ] ) {
153         tx_msg[ 3 ] = arrBUTTON_USB_SCANCODES[ i ];
154         LED1_ON();
155     } //end if arrBUTTON_STATES[ i ] if
156     else {
157         tx_msg[ 3 ] = 0x00;
158     } //end else
159
160     if ( arrBUTTON_STATES[ i ] != arrBUTTON_PREVSTATES[ i ] ) {
161         //usb_put_packet( 1, tx_msg, sizeof( tx_msg ), USB_DTS_TOGGLE );
162         usb_put_packet( 1, tx_msg, 8, USB_DTS_TOGGLE );
163         HID_KEY_DELAY();
164         usb_task();
165         bSentKey = TRUE;
166     } //end if arrBUTTON_STATES[ i ] != arrBUTTON_PREVSTATES[ i ] if
167
168 } //end for
169
170 memcpy( arrBUTTON_PREVSTATES, arrBUTTON_STATES, sizeof( arrBUTTON_STATES ) );
171
172 //debounce keys
173 if ( bSentKey )
174     HID_KEY_DEBOUNCE();
175

```

■ FIGURE 6. The firmware function that processes keyboard hits.

a repeat, we ignore the key until it is released. This instance of how the device is used shows this to be a good feature to have. The de-bounce delay at Line 174 helps to mitigate any transient-caused switching.

Troubleshooting

Troubleshooting this design is pretty straightforward. As with all troubleshooting, we will start with the most likely culprits first:

1) Make sure the device is alive. The firmware is designed so that when the device first powers up our LED will illuminate. As soon as the main loop starts running, that LED will extinguish. Not a fancy check, but it tells us a great deal. It tells us that we have processed all of our initialization code and that the main loop has started. Further, we know that if we get a key hit, the LED will also illuminate and extinguish after our key de-bounce time has expired. So, if our LED doesn't illuminate on power-up, we either don't have power or the power is not within the PIC's range of tolerance. If the LED doesn't illuminate when a key is pressed (key contact shorted), then we need to check the key connections for opens or shorts,

i.e., use a multimeter to check voltages and resistances. There is a possibility that the LED may blink when no keys are pressed. This can occur for two basic reasons: a) The power supply to our device is intermittent. This will cause the PIC to reset and thus re-process our initialization section of code; and b) There could be an intermittent switch closure causing unwanted key presses.

2) Make sure the PC recognizes our device. In most cases (99%+), computers should install the appropriate drivers automatically. Some older PCs (Windows XP SP2 and prior, for example) may not have the correct drivers present. They can be found on the Internet if this occurs.

After the driver installs, open Notepad, simple text, or some other word processing tool. Click some keys (close the key contacts) and you should see the defined scan code key values show up on the screen. If you use the defaults, they will be 0-9, a-d, <return>, and <backspace>. If this doesn't happen, check your device manager area for the USB connected devices, and make sure the device shows up in the list and has a driver installed. If it doesn't, you might need to search the Internet for drivers or check the electrical connection to the USB port/device.

For Windows, there is a simple test application called *usbview.exe* that will neatly show all connected USB devices. There are other tools, but this one usually works fine for Windows. Beyond making sure the drivers install appropriately, I cannot give more detail relative to fixing PC problems without doing a case by case review.

Conclusion

To sum it all up, my friend's video game operates without problems and the project has successfully fulfilled its designed purpose. As a final note, it should be mentioned that this project is exceptionally ideal for any text input program on the computer and also for inputs to satisfy special programs associated with racing event monitoring, weather status, heat/cold or light/dark monitoring, liquid or chemical, or for any other desired input where outside world entry would be the medium.

There are plenty of modifications that can be made to this basic design. Noise and transient voltage suppression of the input switches could be added, should you choose to use the design in a noisy environment such as in an automobile or in an industrial control environment. You could even modify the inputs to be Hall sensor or photoelectric isolated switches.

The firmware could be modified to send words, sentences, or timed outputs based on hold duration, switch closure combinations, and so forth. As long as your changes are within what the PIC documentation states is allowable and you don't violate the USB electrical or communication standards, the possibilities are up to you.

NV

DIY NEVER HAD A BETTER FRIEND...

Take your DIY project to the next level with PanaVise work holding tools. Perfect for hobbies, soldering, electronics, crafting & repair, PanaVise tools have hundreds of uses only limited by your imagination.

PanaVise, "better than a third hand!"

PANAVISE
Innovative Holding Solutions

Now available at
RadioShack

Model 301
640-0178

Model 324
640-0177

www.radioshack.com/StoreLocator

EDMUND SCIENTIFICS®

Wonder and Inspiration Delivered — Since 1942

1-800-818-4955 • ScientificsOnline.com

**\$5
GROUND
SHIPPING**
for N&V Subscribers
USE **SSCTE3**
AT CHECKOUT*

The Brightest Comet of Our Lifetime Arrives This Fall

All eyes this autumn will be on ISON, the potentially record-breaking comet that has astronomers the world over predicting it might be the "Comet of the Century." Coming within 800,000 miles of the Sun's surface in late November, this 'sungrazing' comet should shine as bright as Polaris—or brighter!

These selected Celestron products are ideal for comet-chasing and are guaranteed to maximize your excitement for observation of the Moon, stars, clusters, and nebulae long after ISON has faded away.

Stargaze Anywhere with a Large-Aperture Reflector

A large-aperture reflector design provides a wide field of view, ideal for observing bright comets, galaxies and star clusters. Features a parabolic mirror for the sharpest image and easy, no-tool setup.

- Newtonian Reflector telescope with 114mm (4.49") aperture
- Magnification: 16x-269x
- Finderscope: StarPointer Red Dot Finder
- Adjustable stainless steel tripod
- Eyepieces: 20mm (22x) and 10mm (45x)
- Alt-azimuth mount with panning handle

Celestron 21079 Cometron 114AZ Telescope
#3153327.....\$179.95



Simple Design Makes it Easy for Beginners to Use

Upgraded for 2013, the Cometron FirstScope is a compact, tabletop telescope perfect for beginners looking to experience the magic of a dazzling comet, planet or nebula.

- Dobsonian telescope with 76mm (3") aperture
- Magnification: 15x-30x
- Eyepieces: 10mm and 20mm Kellner

FirstScope 76 Telescope
#3153326.....\$59.95

Track Comets and Find Stars Instantly with This Powerful Binocular

This binocular's massive 70mm objective lenses, multi-coated optics and BaK-4 prisms provide bright, sharp images of comets and celestial objects. Designed for use with your tripod (not included).

- Objective lens diameter: 70mm
- Magnification: 12x
- Includes lens covers, neck strap, tripod adapter & carrying case

Cometron 12 x 70 Binocular
#3153329.....\$89.95



FIND THOUSANDS OF OTHER
UNIQUE SCIENTIFIC ACTIVITIES,
GADGETS, COLLECTIBLES AND MORE.
SCIENTIFICSONLINE.COM

\$5 ground shipping valid on featured products only through 12/15/13

BUILD THE CHROMATICON

By Matthew Bates
mattscottbates@gmail.com

Post comments on
this article and find
any associated files
and/or downloads at
[www.nutsvolts.com/
index.php?/magazine/
article/november2013_Bates](http://www.nutsvolts.com/index.php?/magazine/article/november2013_Bates).



Sometimes discovery occurs during the pursuit of something else entirely. Such is the case of the circuit presented here. I stumbled upon this circuit while trying to design a custom frequency tuner for my guitar. When practicing chords (groups of three note patterns), it's difficult to know when they are being played properly other than how they sound. There's no electronic device that I'm aware of that can measure if the chords are being played correctly. If I try to use a standard frequency tuning device, a "C" chord displays as the "G" note. So, I ended up building a circuit that would respond to what sounded correct, and that turned out to be the discrimination of the entire octave the chord was played in. This led to the construction of a device that could respond to any one of the seven traditional octaves. I found that playing music through the device was like having my very own extremely responsive computerized light show. I can truly say without equivocation that, "This circuit is hands down the best sound to light device you could ever build."

What Makes This Device So Different

The CHROMATICON is actually an octave display device and should not be confused with circuits that are often referred to as color organs. Traditionally, color organ circuits that translate sound to light are active bandpass filters comprised of capacitors, resistors, and op-amps. These filters separate the low, mid-range, and high frequencies that can be used to illuminate different wavelength LEDs.

The last color organ circuit I tried building was purchased after watching a slick YouTube video entitled "Circuit Skills." The video was for a kit bundle that sold for around \$40 and contained quite a few parts. For some reason, having so many parts made me feel like it had to be better than the last one I built. After taking what seemed like forever to construct and

tweak, its performance fell short of what must have been too great an expectation on my part.

Just like other similar circuits I have built in the past, I find that the wholesale summing of frequencies in this manner to be somewhat uninteresting. Personalizing these circuits typically means just selecting a different color combination of lights for the three bands, and scalability is limited to the strength and number of lights. This is where the CHROMATICON and the color organ circuits part company.

With color organ circuits, the patterns that can be displayed are fixed by the physical placement of the LEDs. With the CHROMATICON, however, displayed patterns are controlled by software. Instead of a sluggish response to the music input, the CHROMATICON comes to life with each note played, and every song displays in some unique way. It has caused me to have a renewed interest in my collection of music, and I am anxious to see how each song will get displayed.

What Exactly Are Octaves

The following is a simple overview of the fundamentals of how music is composed, and may help in understanding the approach I have taken toward my technique of sound to light conversion. Traditional music is comprised of individual frequencies that make up the seven note or *diatonic* scale consisting of the notes C, D, E, F, G, A, and B. If you include the sharps (notes raised a half tone), there are actually 12 notes to the *chromatic* scale written as C, C#, D, D#, E, F, F#, G, G#, A, A#, and B.

On a piano, these 12 note scales repeat as the first octave (CDEFGAB) to seventh octave (CDEFGAB), plus a minor third (A0 to C8) across a keyboard of 88 keys. An octave is the simplest interval in music. It is the interval between notes in one octave to the same note in a lower or higher octave. The range of frequency between notes from one octave to the next is either one half or twice the frequency of the note.

For example, the A note from the fourth octave has a frequency of 440 Hz. The A note one octave higher (A5) is at 880 Hz, and the A note an octave lower (A3) is at 220 Hz. With the octave being such a fundamental part of creating music, it just makes sense to me to build a circuit that can visualize the notes played in the full range of octaves used to compose music.

Digital Hardware for an Analog Task

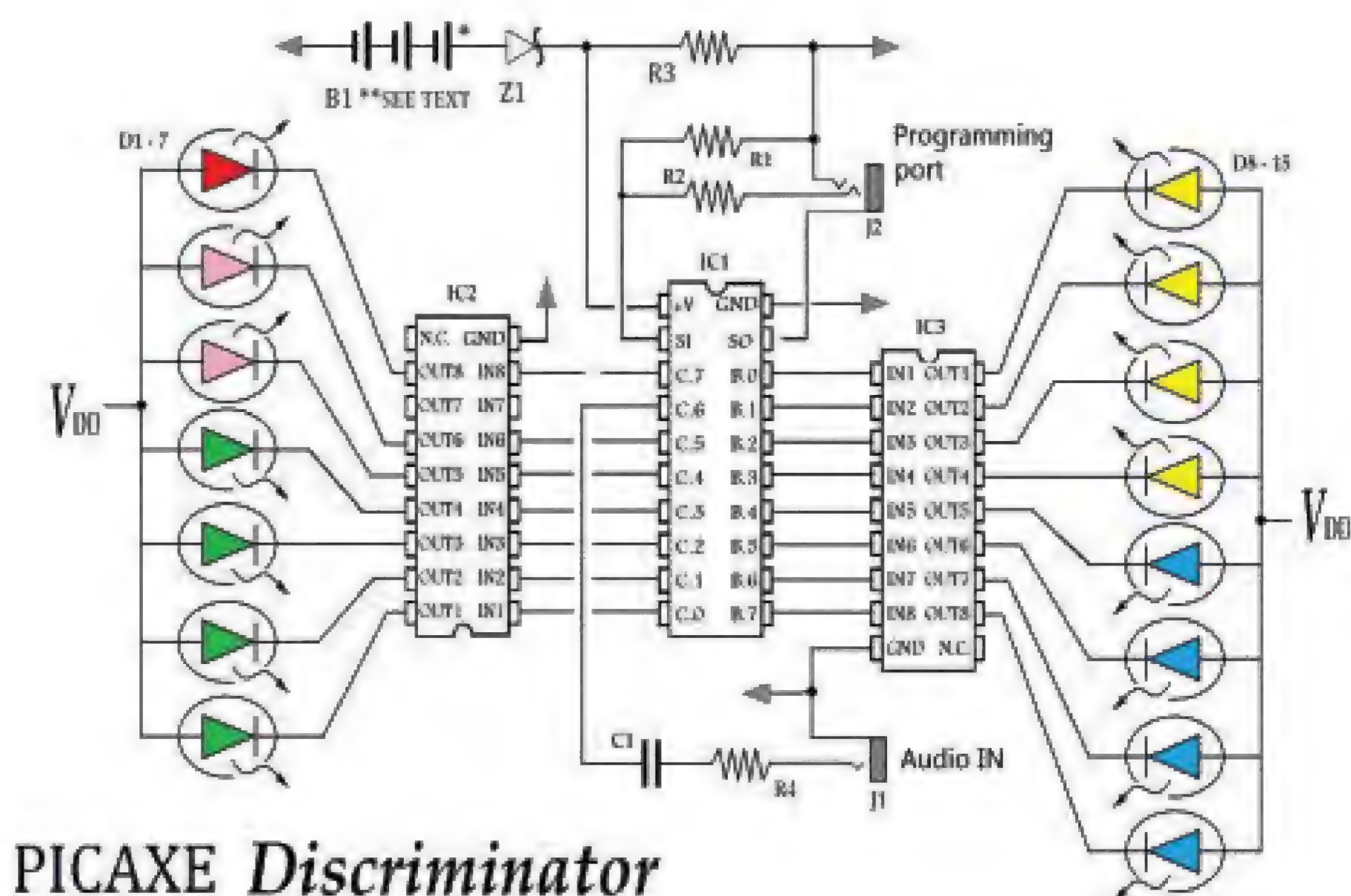
The heart of this circuit is the PICAXE, but it can be constructed using any processor that is capable of frequency measurement. The PIC, PICAXE, and Stamp micros are able to measure frequency with the *COUNT* or *PULSIN* command. I haven't tested the circuit with the Atmel family (Arduino), but I believe that it does have a similar *Pulse(IN)* command.

The operation of the circuit is extremely straightforward. A line level output from an audio source is coupled to an input pin on the 08M2 by way of a 0.1 μ F capacitor through a limiting resistor. The PIC 20M2 chosen for this project allows for up to 15 outputs to drive the LEDs, and one pin necessary for the audio input.

There are two design camps when using this method to convert sound to light. In one camp, by hard-wiring patterns of LEDs more patterns can be had using fewer I/O ports. An alternative method is to individualize the connections to the LEDs. This method will exhaust I/O lines faster, but allows for greater flexibility in displaying patterns of LEDs by simply changing the software. This is how the circuit covered here was designed.

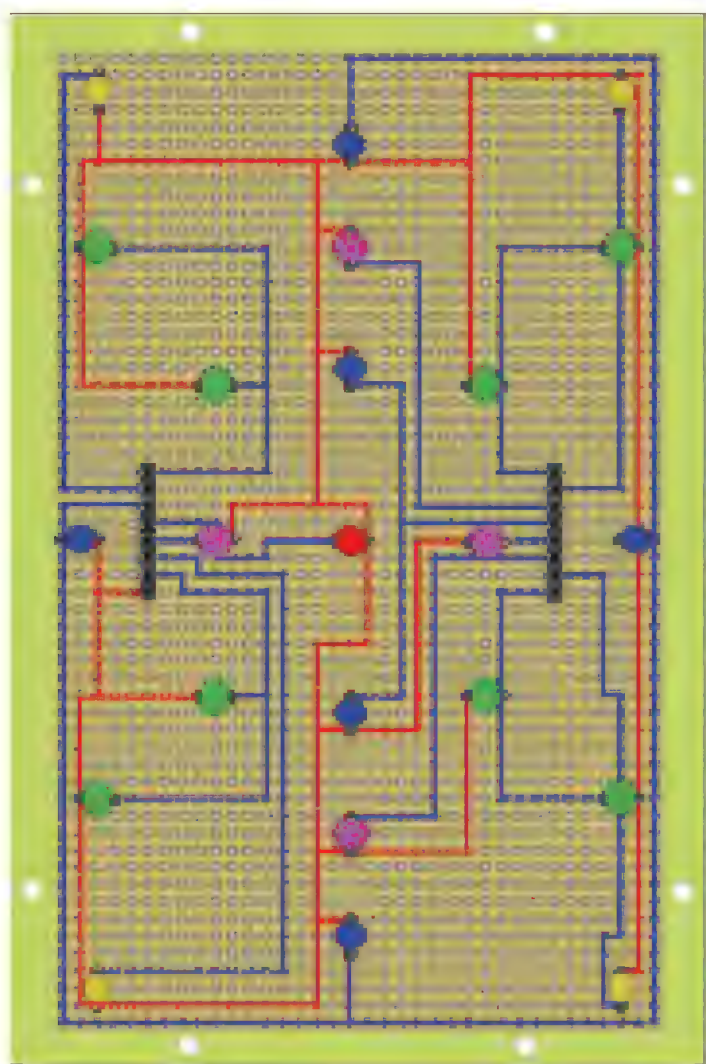
Following the schematic shown in **Figure 1**, a left or right channel line out from an audio source is coupled to the C.6 pin of the PIC using a 1 μ F capacitor. The only difference between this circuit and the minimal circuit is the use of an additional Darlington array to meet the higher I/O count.

Any medium output LED will work for the display. The medium output range is usually between 2,500 and 8,000

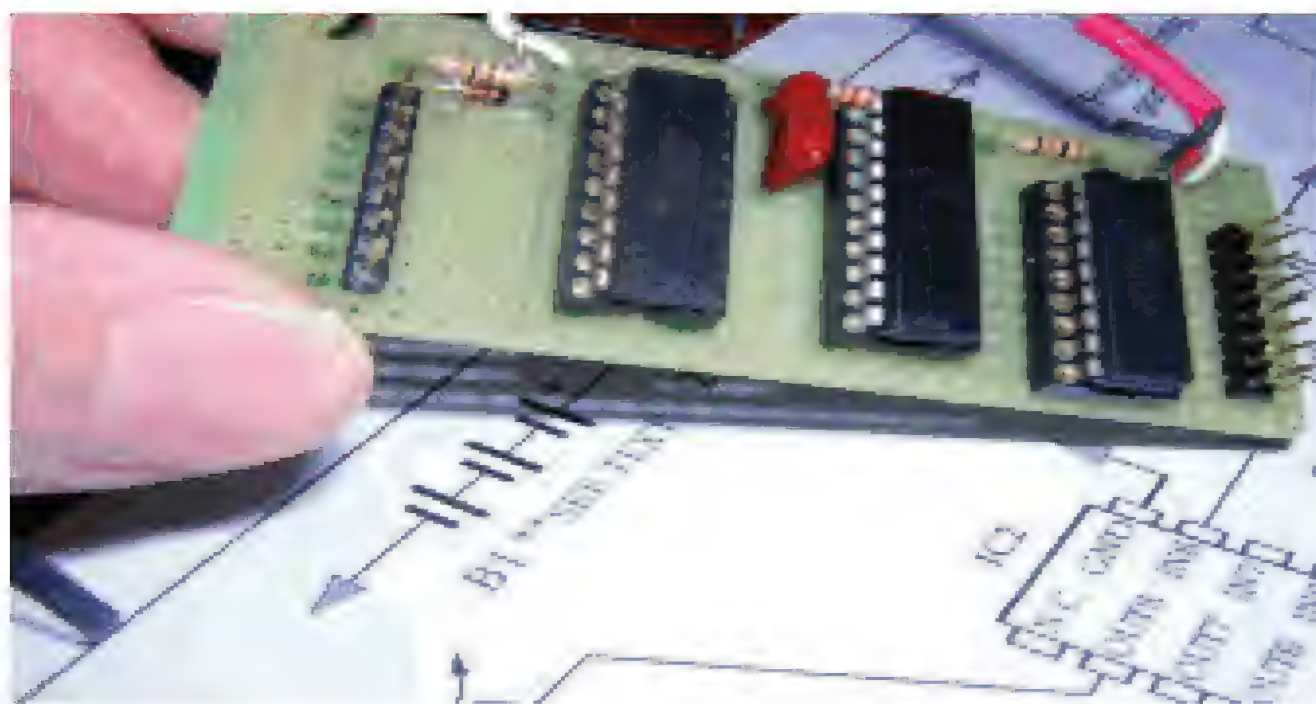
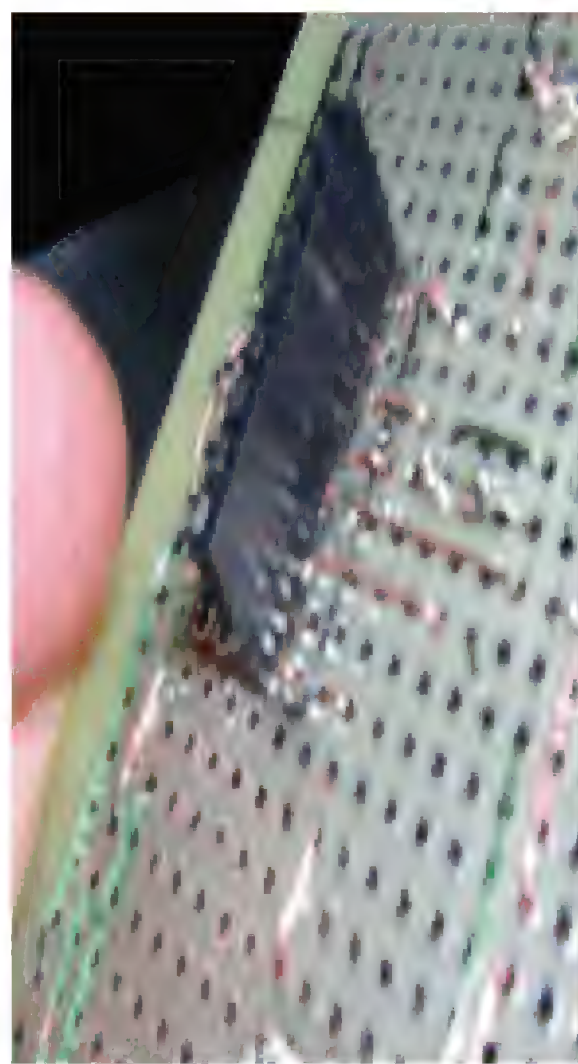


PICAXE Discriminator

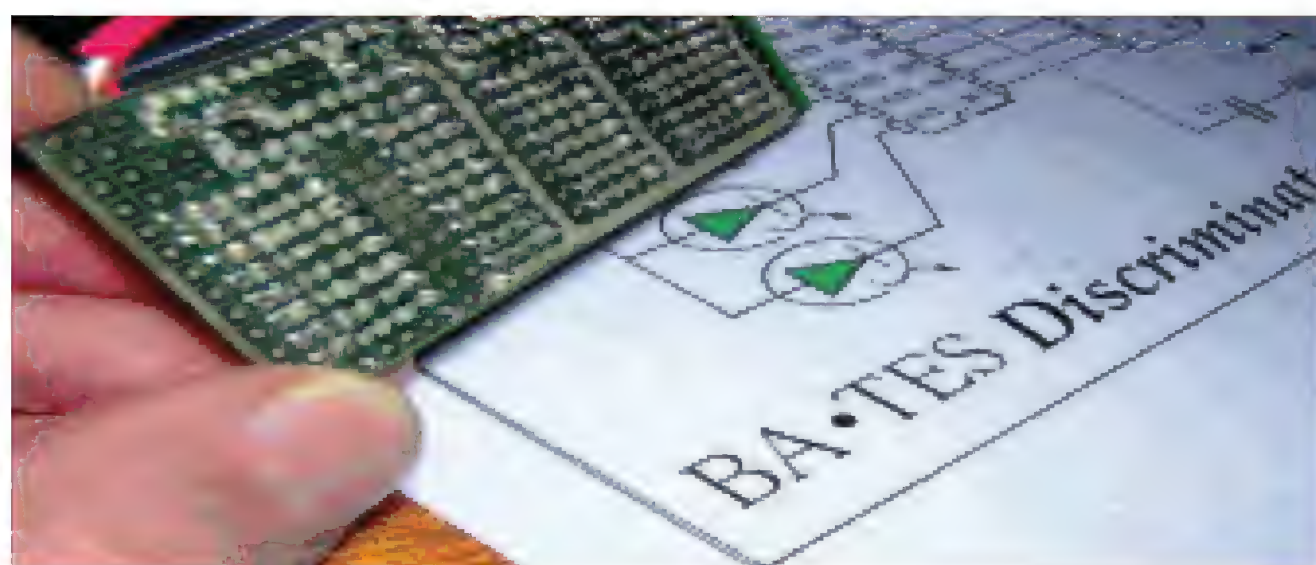
■ **FIGURE 1. Example system design using the PIC 20M2 micro-controller.**



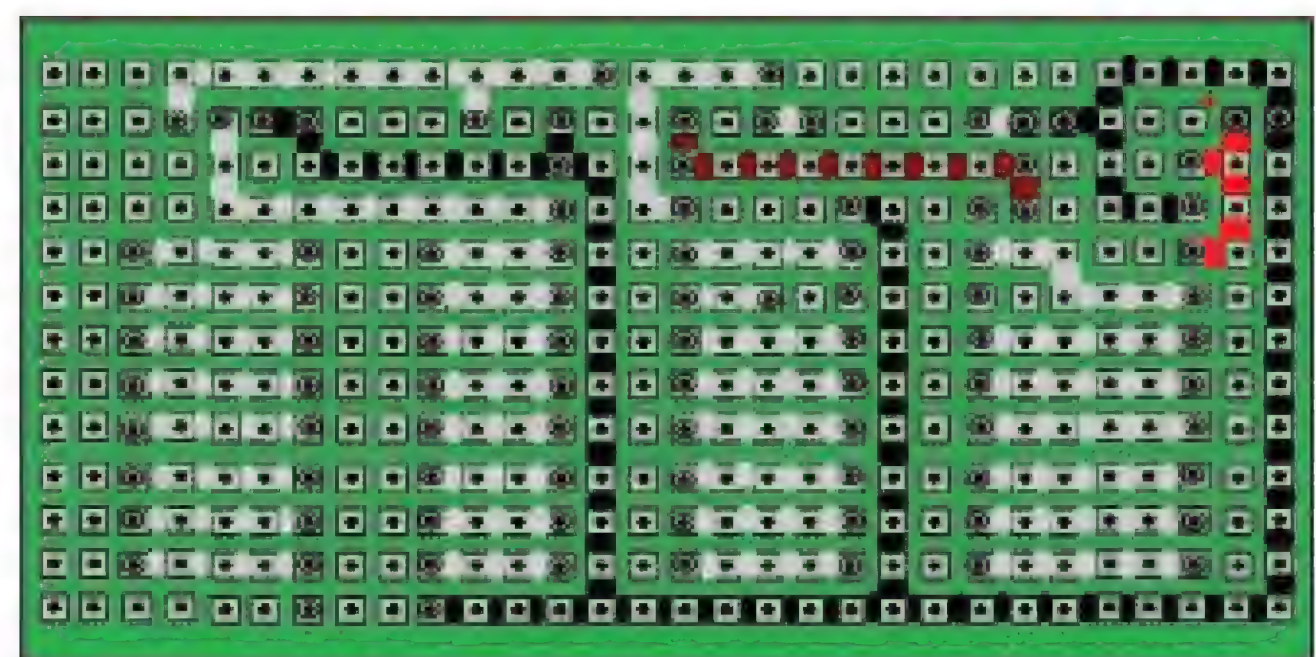
■ **FIGURE 2.** Interconnect for the LEDs and a close-up of the display wiring.



■ **FIGURE 3.** Component side of the driver circuit.



■ **FIGURE 4.** The wiring side of the driver board.



LEGEND: ■■■■ + 2.7v ■■■■ GND ■■■■ +5v ■■■■ SIGNAL

mcd. LEDs used in this design are in that range, and all have a maximum current rating of about 22 mA. The LEDs are pulsed and are never turned “on” for longer than 50 msec. This keeps current down to around 15 mA and allows you to parallel diodes without the need for current-limiting resistors.

You may want to include resistors while experimenting, and they should be added between the outputs of the Darlington transistors and the cathode side of the LEDs. Be sure to use the LINE OUT from the audio source and not the amplified output. There is a minimum voltage threshold for the input (approx 1V peak to peak). It’s best to start at the maximum LINE OUT setting from the source device and then lower it as needed for personal taste.

Building the Display Board

Figure 2 is an example layout of LEDs for the CHROMATICON. The board used for the display is a 4” x 6” plated through hole prototyping board. With 15 I/O lines available, there are many options for the organization of LEDs. If you connect more than one LED per line, connect the additional lights in parallel. The individual and grouped diodes are wired to female SIP headers of eight pins each on either side of the board, mounted in the center as shown in **Figure 2**.

The only pin that matters for alignment to the driver circuit is the +5V supply line. The remaining connections to the LEDs do not need to follow any strict order and should be made opportunistically trying to avoid using jumper wires.

Building the Driver Board

Figure 3 is a photo of the driver circuit that was used in the prototype. The driver contains all of the electronics needed for the circuit. The components on this board were connected together using silver conductive ink, but any interconnect method will work. **Figure 4** is the back or interconnect side of the driver board. The connections for the prototype were made with conductive ink but any method of connection can be used. The parts are very close together, making it easier for interconnect.

Figure 5 shows the reflected inking pattern for the design. The connections from the back need to be reflected to insure that the pin 1 locations for the ICs are correctly connected. This same pattern can alternatively be used for soldering wire busses to connect the components. The routes are color coded to separate power, ground, and signals. **Figure 6** shows a side view of the board with the correct placement of the male headers. They need to be mounted so the pins are facing the direction shown to plug into the female headers on the display board.

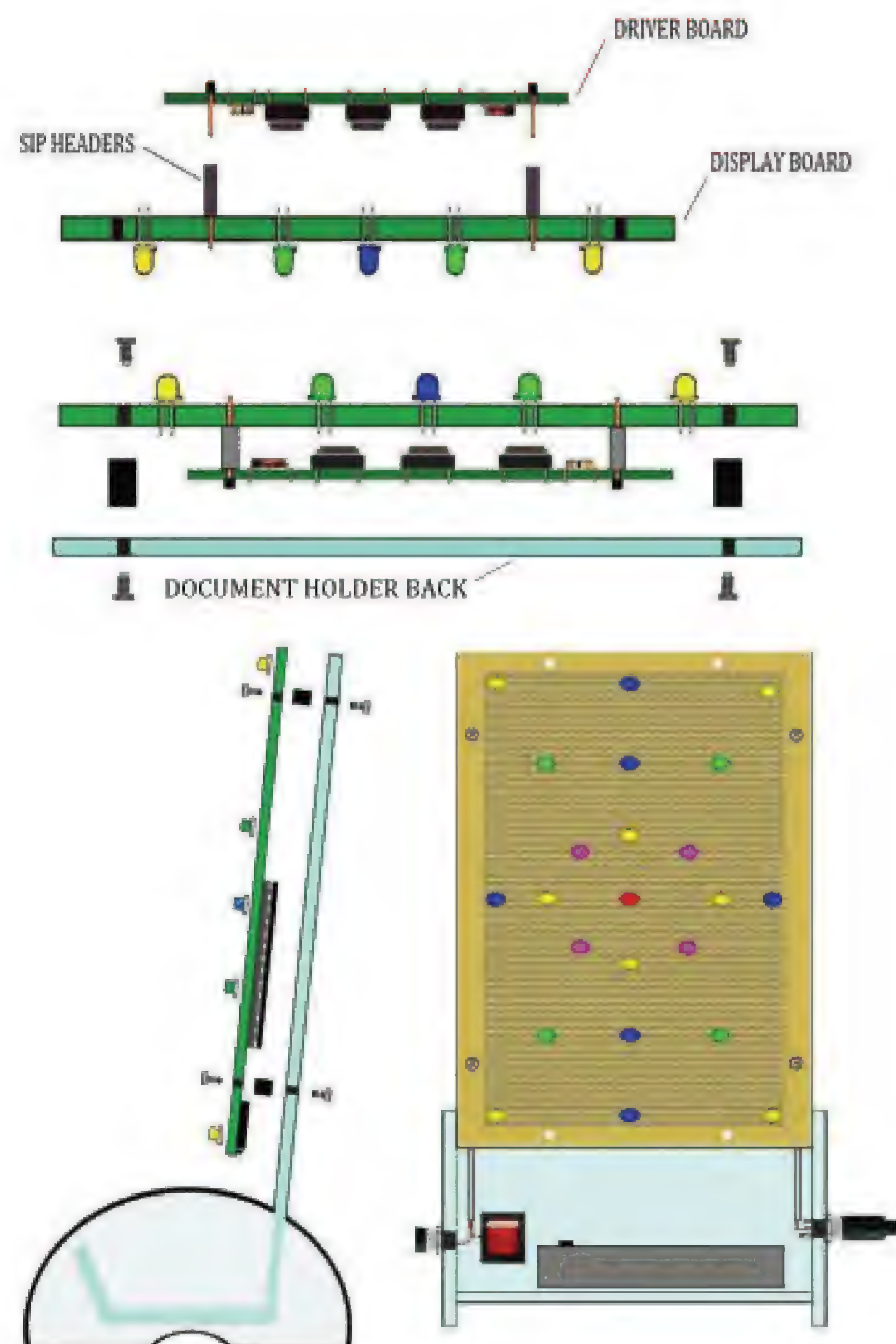
Wiring for the circuit is brought to the base of the document holder behind the display board. This consists of the programming cable, audio input, and power supply for the circuit as shown in **Figure 6**. You may want to use a shielded

■ **FIGURE 5.** Connection side of the driver board.

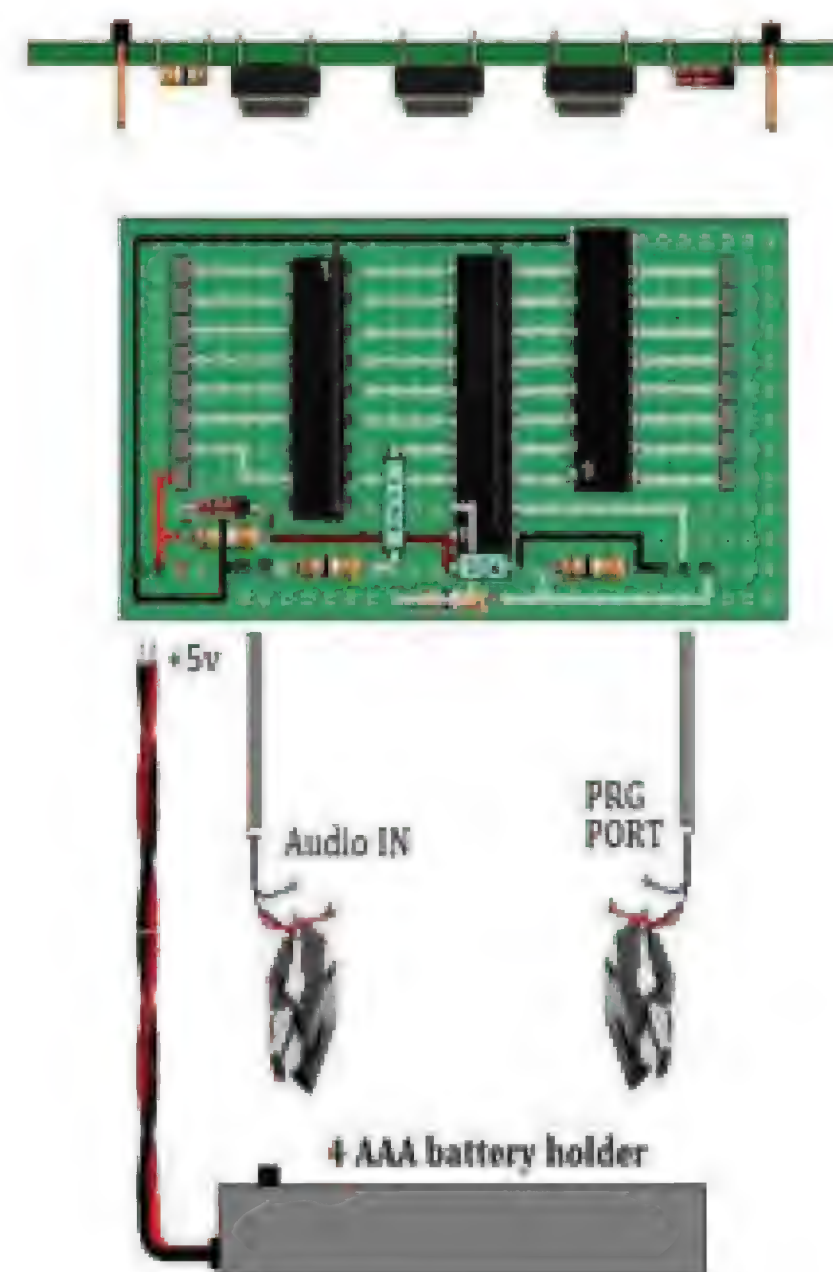
cable for the connection to the 3.5 mm audio jack for the Audio IN. Connection of the programming port requires a specific connection to the jack as described in the PICAXE manual. The four AAA battery pack is connected directly to the board and doubles as the power switch for the circuit. If you like, you can substitute or add a 5V supply such as from a USB port or other rectified source to power the circuit without modification to the electronics. The NiMH supply will last about a week before needing to be recharged.

Figure 7 is an example of the assembled components. The display driver connects to the back of the display board in a shield-like fashion using SIP headers. With the driver connected to the display board, the entire assembly mounts to the upright panel of the document holder, also shown in **Figure 7**. The two eight-pin male headers located on the driver board supply the display board with 15 I/O lines and a positive supply by way of the two headers. Threaded standoffs are used to attach the display and driver shield to the frame of the document holder.

The Discriminator Software



■ **FIGURE 7.** Display and driver board connecting to the front of the document holder.



■ **FIGURE 6.** Wiring connections to the driver board.

Software for the circuit is pretty straightforward. We are basically using the PIC as a low-level resolution frequency counter. I

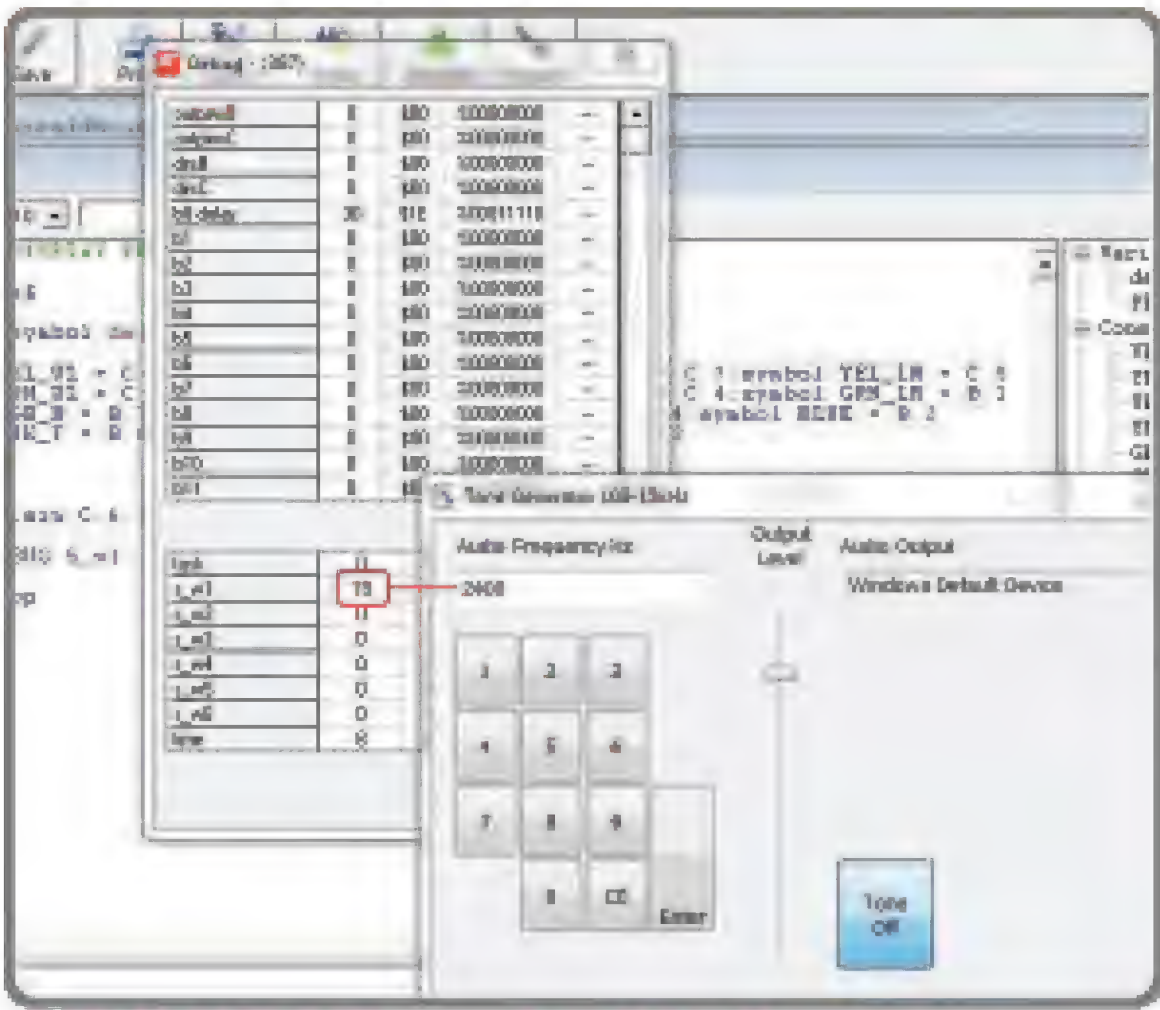
began writing this software using the *COUNT* command to actually measure the literal frequency. This would work fine if we were just measuring static tones, but for dynamic music it turned out to be a little too slow. The alternative is to use the *PULSIN* command which measures the length of the incoming pulses and produces a corresponding number between zero and 255. This is less graceful but at the end of the octave sampling day, achieves the same net result. You just have to be willing to translate your target frequency range initially into their corresponding 256 bit values, and then use that translated value in your conditional statements. This is actually easier than it sounds. To translate the output using the *PULSIN* command, start a DEBUG window and input a frequency to the PIC using the following example code:

```
DO ; begin loop
PULSIN C.6, 1, W0

; measure the frequency on pin C.6 and put
; result in variable W0
DEBUG W0 ; start DEBUG monitor
LOOP ; do again
```

The DEBUG window will show a list of all your variables. If the input frequency was 2,400 Hz, the translated value of 19 is what got stored in W0 as shown in **Figure 8**. Instead of using the actual frequency in Hz for the octave range, you would record the translated value for both the low and high end of the octave range and then use those values in your conditional statements.

The software that accompanies this project already



■ FIGURE 8. Example of the frequency translation process.

has translated values. You may want to double-check the calibration, especially if you make any modifications to the supplied schematics. The *PULSIN* frequency measurement command requires only three arguments which are the port or *Pin#* for the input, the *State* from which to measure from, and the *Variable* used to hold the value for the frequency. This command measures the length of a pulse on the input pin. Depending on the *State* variable, it either measures from the rising edge of the input or falling edge indicated by declaring a 0 or 1 value on the command line. The measured result is placed in the variable of choice. For example:

```
PULSIN C.6, 1, W0
; measure the low to high transitions on C.6
; & stores in W0
```

ITEM	QTY	DESCRIPTION	SOURCE/PART#
BRD1	1	Proto Board	Digi-Key V2009-ND
BRD1	1	Proto Board	Digi-Key V2025-ND
SCKT1	1	20 Pin Socket	Digi-Key A106188-ND
SCKT2	2	18 Pin Socket	Digi-Key ED3019-ND
IC2,3	2	ULN2803	Digi-Key ULN2803APG(OHZA-ND)
J1,2	2	3.5mm Audio	Digi-Key CP1-3543-ND
Z1	1	2.7V Zener Diode	Digi-Key 1N5223B-ND
C2	2	.01 μ F Capacitor	Digi-Key P4525TB-ND
R1	1	10K 1/4W Resistor	Digi-Key A105970CT-ND
R2	1	22K 1/4W Resistor	Digi-Key 22KAACT-ND
R3	1	100 Ohm Resistor	Digi-Key 10QBK-ND
R4	1	10 Ohm Resistor	Digi-Key 100DSCT-ND
SP	4	1/2" Standoffs	Digi-Key 3480K-ND
PH	8	4-40 Screws	Digi-Key 335-1082-ND
D1-8	8	Green LED	Digi-Key C4SMF-GJS-CV0Y0792TB-ND
D9-14	6	Blue LED	Digi-Key C503B-BCS-CV0Z0461-ND
D15-18	4	Yellow LED	Digi-Key 897-1071-ND
D19-22	4	Pink LED	Digi-Key 67-2062-ND
D23	1	Red LED	Digi-Key C503B-RBN-CW0Z0AA1-ND
HDR1	1	40 Pin Male SIP	SparkFun PRT-00116
HDR2	1	40 Pin Female SIP	SparkFun PRT-00115
IC1	1	PICAXE 20M2	SparkFun COM-10807
BATT	1	4AAA Batt Holder	RadioShack Cat# 270-411
HLDR	1	Document Holder	Office Max SKU# 309264
MISC		Shielded 2 conductor wires for audio connection.	

Note	Frequency (Hz)
C ₄	261.63
C# ₄ /D ^b ₄	277.18
D ₄	293.66
D# ₄ /E ^b ₄	311.13
E ₄	329.63
F ₄	349.23
F# ₄ /G ^b ₄	369.99
G ₄	392.00
G# ₄ /A ^b ₄	415.30
A ₄	440.00
A# ₄ /B ^b ₄	466.16
B ₄	493.88

Table 1. Frequencies of notes for the 4th Octave.

The default 4 MHz clock speed for the microcontroller is more than adequate for measuring audible frequencies. By using a table of the established frequencies for musical notes, the result held in the *W0* variable can be used to initiate some event such as lighting a designated LED or group of LEDs. We can use a list of the frequencies that make up the chromatic scale for any octave, such as the example shown in **Table 1** used for the fourth octave scale. Using this list and with the help of a tone generator, you can compile an inventory of the translated numbers for each octave. For the

list in **Table 1**, the translation would look something similar to this:

```
C4 = 261 Hertz   Translated value = 17
                    (beginning of the 4th Octave)
B4 = 493 Hertz   Translated value = 37
                    (end of the 4th Octave)
```

To light an LED or group of LEDs with a specific wavelength that includes a range of frequencies, you only need to provide a conditional *IF/THEN* statement for the range desired. The following is a code fragment that would cause the grouping of four yellow LEDs shown in the **schematic** to light if the sampled input frequency was between 260 and 493 Hz:

EXAMPLE DISCRIMINATION FOR THE 4TH OCTAVE :

```
If W0 > 16 and W0 < 38 THEN
; any notes between 260Hz
; to 494Hz

HIGH Y_UL,Y_UR,Y_LL,Y_LR : PAUSE
DELAY:LOW Y_UL,Y_UR,Y_LL,Y_LR
; light these 4 LEDS
```

This *IF/THEN* statement lights the four LEDs connected to ports B.0-B.3 if any notes within the fourth octave are sampled. You can go even further to enhance the effect of the translation of sound to light by carving up the octave into several patterns of yellow lighting. We can accomplish this by lighting the upper left (*Y_UL*) and lower right (*Y_LR*) LED if it is the lowest part of the octave, and upper right and lower left LED if it is the highest frequency part.

By reserving just the center part of

the octave to light all four lights, it creates a very interesting effect. It helps to assign names for the ports that is indicative of their physical locations. This same technique can be used to indicate when notes from the other octaves are played.

Using LEDs of the same color arranged in various patterns creates another interesting effect. I find that by using enough LEDs to make square, diagonal or cross patterns can produce the best effects with the least number of LEDs. The layout of LEDs shown in **Figure 2** is an example of this suggested arrangement. The software that is available for download was written expressly for the arrangement of diodes shown in **Figure 5**. The code has been well documented and should serve as a starting point for a layout of your own design. This design can be scaled up using several methods. One method to free up I/O lines would be to hard wire the LED patterns. This would allow you to have more patterns. Another method would be to connect more LEDs in parallel to the existing I/O lines. Each ULN2803 is rated up to 500 mA.

The duration for how long the LED is kept open during operation is controlled by the value held in the *DELAY* variable. In the program listing provided, the *DELAY* is set to 50 milliseconds, but you may want to use

different delay times for the ranges of the discriminated frequencies. There can be a noticeable difference using longer on times for higher frequencies, and shorter times for the lower frequencies.

Final Thoughts

This circuit is a single channel or mono design, so the choice of using the left or right channel is up to you. A stereo circuit may require more than just another input due to the time needed for both sampling and processing the results. A twin circuit would be ideal to produce a stereo effect. A sensitive microphone can be used as an input to the processor, as long as an amplifier is included to boost the signal. An alternative to using a separate board for the driver would be to mount the components directly to the back side of the display board. The 4" x 6" board for this project is a plated through hole design which would allow the parts to mount on either side. If you prefer that kind of design, feel free to email me if you need help with that implementation.

However you decide to implement this circuit, I hope you enjoy how it enhances the music you listen to as much as I have. **NV**

Images Scientific Instruments Inc.

PIC Basic Project Board



16F88 PIC Basic Project Board Features

- LCD Display - backlight & contrast control
- 2 A/D channels,
- 8 digital I/O lines
- 5V and 9V operation
- Free Student version of PICBASIC PRO & Microcode Studio
- 16F88 Programming Service Available
- Project library with source codes available

Sensors:	Do More:
Humidity	Frequency Meter
Temperature	Pulse Generator
Toxic Gas	Elapse Timer
Bend and Force Sensors	Radiation Pulse Counter
CdS Light sensors	Volt Meter

www.imagesco.com/microcontroller/16f88.html

The Convenient All-in-One Solution

for Custom-Designed Front Panels & Enclosures



FREE Software

➔



ONLY \$90.24
with custom
logo engraving

You design it
to your specifications using
our FREE CAD software,
Front Panel Designer

We machine it
and ship to you a
professionally finished product,
no minimum quantity required

- Cost effective prototypes and production runs with no setup charges
- Powder-coated and anodized finishes in various colors
- Select from aluminum, acrylic or provide your own material
- Standard lead time in 5 days or express manufacturing in 3 or 1 days



FrontPanelExpress.com
1(800)FPE-9060

BUILD A PERSONAL GUITAR MICROPHONE PRE-AMP

By Ron Hoffman
RJHEE@aol.com

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?magazine/article/november2013_Hoffman.

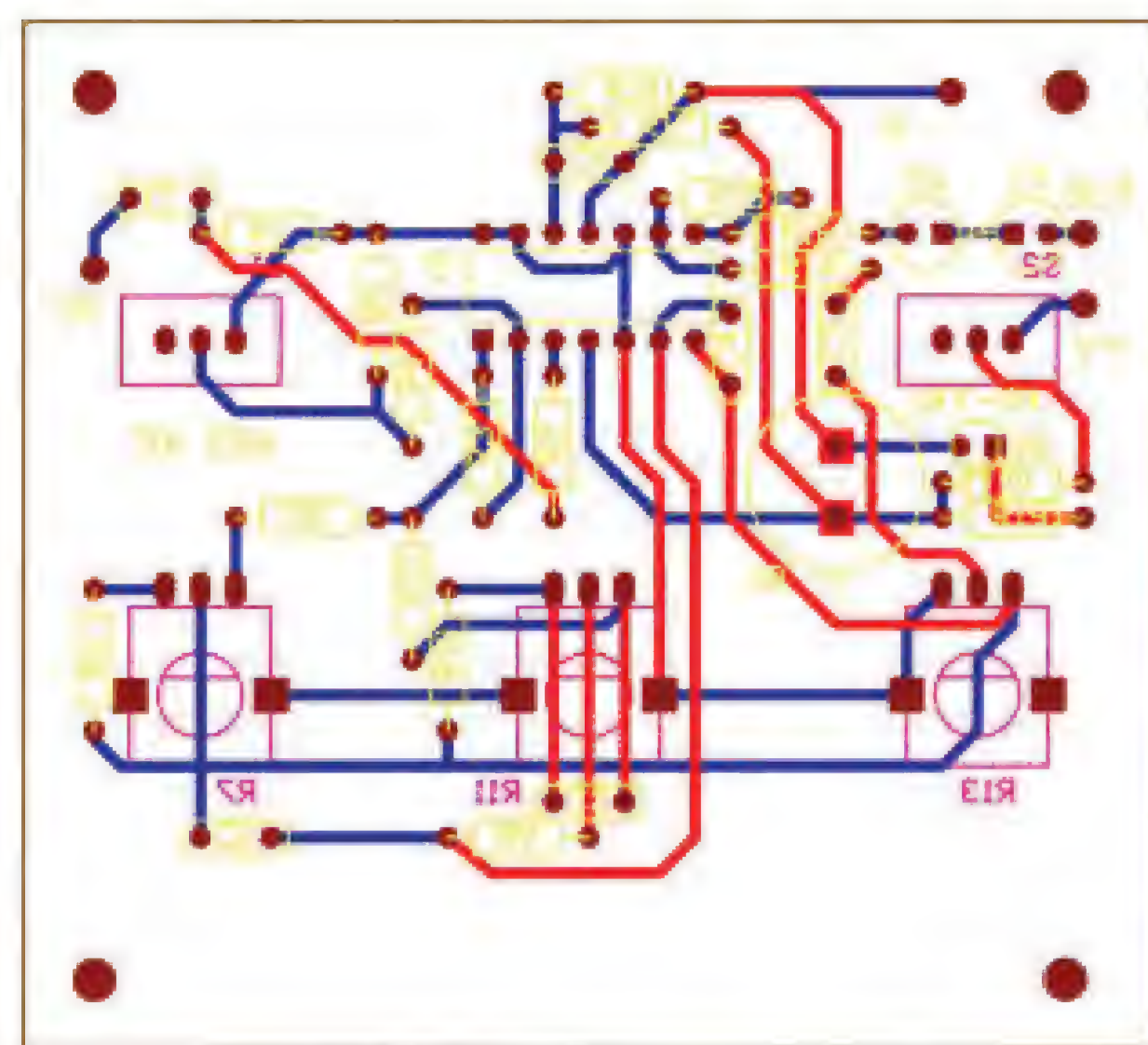


Most projects arise out of the need to make something that isn't available, or at least isn't available at a reasonable cost. I have seen many four-channel pre-amp circuits that offer adjustable gain, but not an equalizer. If you want an equalizer, then you are basically talking about a mixer (expensive), multi-channel, effects, etc. I wanted just a simple guitar/microphone pre-amp with treble, bass, and volume adjustment. "Why?" you ask. Well, I built and installed an electret microphone in my Ibanez acoustic guitar which I have been playing for more than 35 years. It consists of an electret microphone, a bias resistor, a blocking capacitor, two AAA batteries, an on/off switch, and a 1/4" phone jack mounted through a hole in the side of a guitar.

When I play through my mixer or guitar amp, I adjust the tone controls to give the guitar a rich sound that is balanced — not too tinny, trebly, or boomy bassy sounding. I play at many different open mic nights at various clubs, bars, and restaurants in the area. They all have house systems, but don't like to change their settings to accommodate "guests." A lot of the miked guitars have built-in pick-ups and pre-amps, and they can adjust the sound on their instrument. I couldn't ... until now.

I designed and built the GMPA-4 (Guitar Mic Pre-Amp, version 4) to allow anyone to connect to an amplifier or house PA system, and adjust their microphone or guitar pick-up for the proper tonal sound and volume that they desire. The circuit is version 4 because of tweaks I made to its versatility, keeping the cost low, and giving good battery life.

I tested the GMPA-4 with my Ibanez acoustic guitar with built-in electret mic and my Les Paul electric guitar with dual hum-bucking pick-ups. It worked flawlessly.



■ FIGURE 2. PCB layout.

The pick-ups put out several hundred millivolts and act like a miniature generator when the steel string vibrates across the Alnico pole magnet under each string.

The GMPA-4 input impedance is roughly 100K ohms which does not load down the guitar output. I also tried using a Nady SP-4 microphone with a high impedance matching transformer. It worked perfectly, as well. If you use a low impedance microphone, you need to use a balanced low impedance to unbalanced high impedance matching transformer.

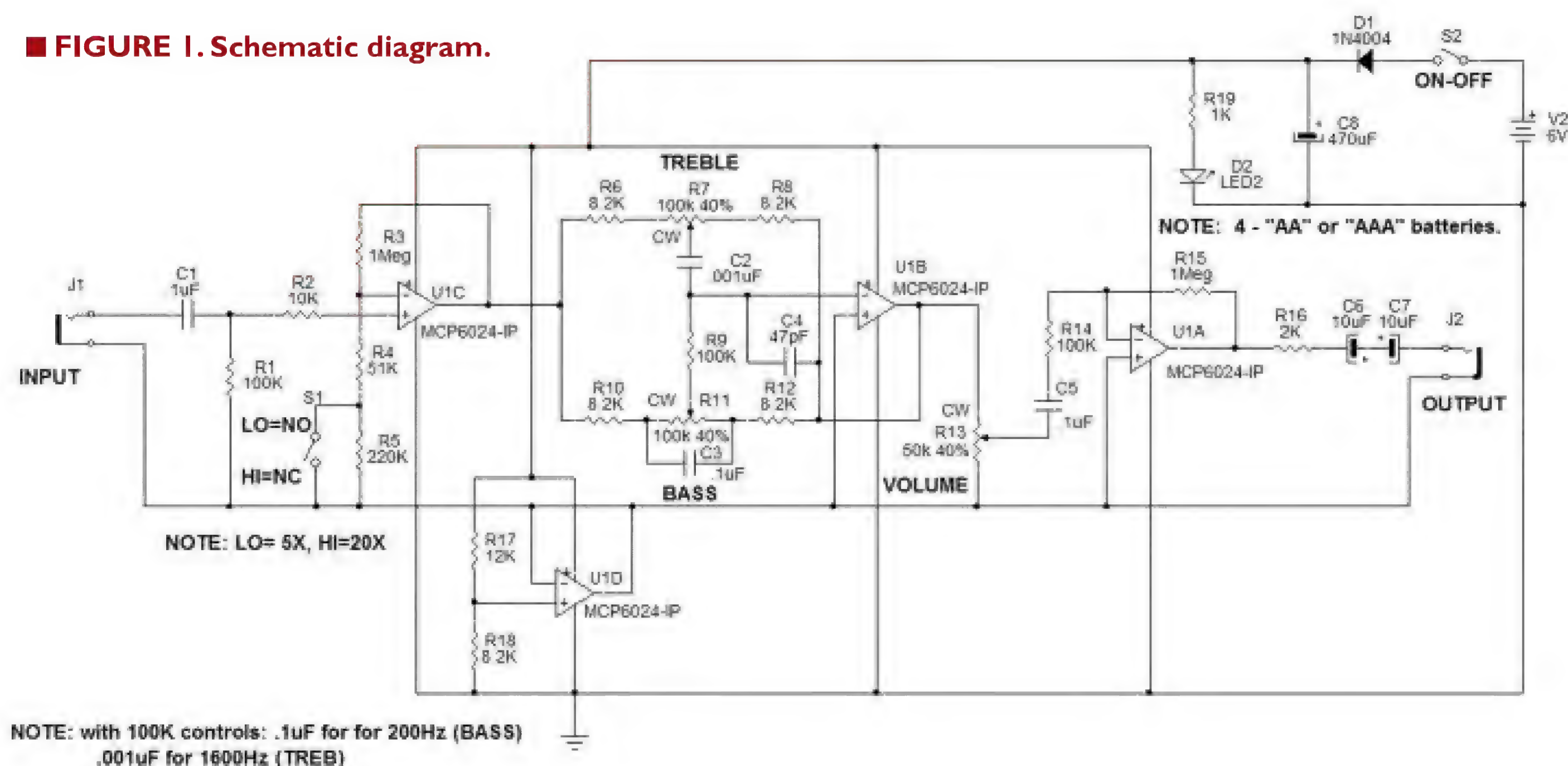
It would be possible to add the impedance matching transformer and XLR balanced line jack inside the unit, but this was not done in order to save cost and space. It would also be possible to add additional channels by adding additional quad op-amps and the necessary components.

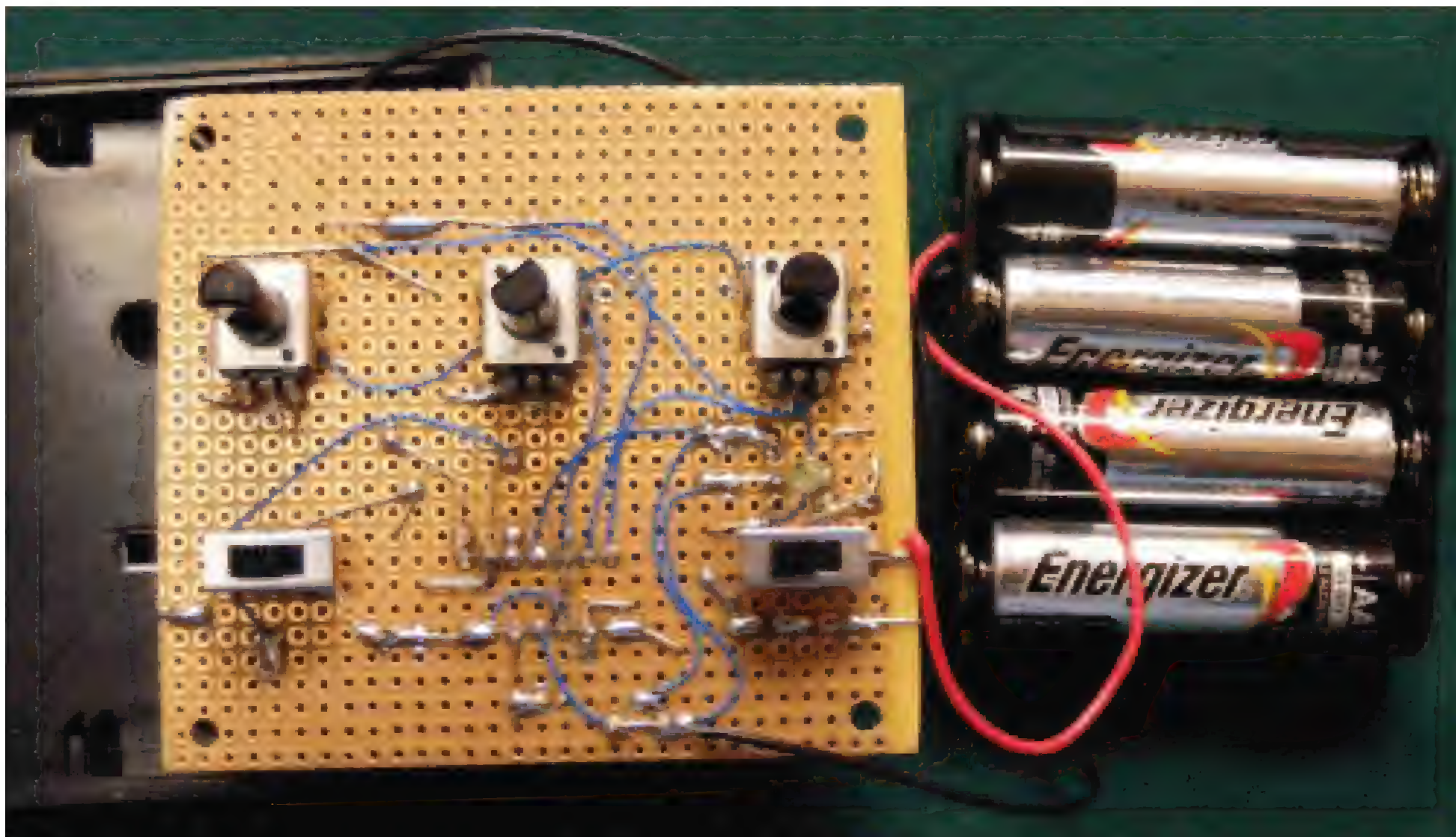
At most venues I play, they have microphones already set up, so I only need the one channel for my guitar. I kind of like having the one unit per instrument so each person can select their own levels and tonal settings.

About the Circuit

The schematic diagram (Figure 1) shows the MCP6024 quad op-amp which is the heart of the unit. U1C is the input buffer and has two gain levels that are selected by the HI-LO switch. When the switch is open, the gain is about five. When the switch is closed, the gain is about 20. The output of U1C feeds the +12 dB side of the treble and bass controls; R7 and R11, respectively. The signal

■ FIGURE 1. Schematic diagram.





■ **FIGURE 3. Control side of PCB.**

feeds the negative input of U1B. The output of U1B is fed back to the treble and bass controls, and to the volume control R13 which has an audio taper.

The signal is coupled through C5 and R14 into the negative input of U1A. The double inversion maintains the proper phase from input to output through the signal chain. The fixed gain of the output stage is 10. Thus, the user has a choice of either 0 to 50 or 0 to 200 signal gain, depending on the input signal level and the output

signal level desired.

The output of U1A is fed through R16, C6, and C7 to the output jack. The MCP6024 is an RRO (rail-to-rail) output op-amp. The input is linear from 0 to Vdd - 1.2 V. By using four AA batteries (six volts) and a reverse polarity protection diode D1 (-.6 volts), the supply to the MCP6024 is 5.4 volts. The op-amp inputs can go as high as 4.2 volts in the linear range.

To maximize the voltage swing, the amplifier bias level is set by R17, R18, and U1D. The values selected give a reference voltage of 2.2 volts which is roughly half of the linear range.

The output of U1D is the ground

reference for the unit and is connected to the input/output jack grounds, as well as the positive input reference for the three other op-amps in the signal chain.

The green LED shows if the power is on and the relative health of the batteries. The shelf frequencies for the treble and bass controls are 1.6 kHz and 200 Hz, respectively.

This gives more adjustment (about ± 1.5 dB) of the sound's brightness to the treble control.

QTY	REF#	SOURCE	DESCRIPTION/VALUE
3	C1,C3,C5	DK	.1 μ F metalized poly, $\pm 10\%$, Panasonic ECQ-V1H104JL, Digi-Key P4525-ND, or equiv
1	C2	DK	.001 μ F metalized poly, $\pm 10\%$, Panasonic ECQ-B1H102JF, Digi-Key P4551-ND, or equiv
1 (Opt)	C4	DK	47 pF ceramic cap
2	C6,C7	DK	10 μ F alum elec, 35V, Panasonic ECA-1CM331, Digi-Key P5161-ND, or equiv
1	C8	DK	470 μ F alum elec, 16V, Panasonic ECA-1CM471, Digi-Key P5141-ND, or equiv
1	D1	DK	1N4005, 1A, 400V, diode, Fairchild 1N4005FSCT-ND, or equiv
1	D2	DK	LED, GRN (green), Lite-On LTL-4232N, or equiv
3	R1,R9,R14	DK	100K resistor 5%, YAGEO 100KQBK-ND, or equiv
1	R2	DK	10K resistor 5%, YAGEO 10KQBK-ND, or equiv
2	R3,R15	DK	1 meg resistor 5%, YAGEO 1.0MQBK-ND, or equiv
1	R4	DK	51K resistor 5%, YAGEO 51KQBK-ND, or equiv
1	R5	DK	220K resistor 5%, YAGEO 220KQBK-ND, or equiv
5	R6,R8,R10, R12,R18	DK	8.2K resistor 5%, YAGEO 8.2KQBK-ND, or equiv
1	R16	DK	2K resistor 5%, YAGEO 2KQBK-ND, or equiv
1	R17	DK	12K resistor 5%, YAGEO 12KQBK-ND, or equiv
1	R19	DK	1K resistor 5%, YAGEO 1KQBK-ND, or equiv
2	R7,R11	DK	100K linear pot, BOURNS PTV09A-4225F-B104-ND DK
1	R13	DK	50K audio pot, BOURNS PTV09A-4225F-A503-ND
2	S1,S2	DK	SPDT miniature slide switch, STS121PC04 450-1609-ND
1	U1	DK	MCP6024-I/P quad RR op-amp, Microchip Tech MCP6024-I/P-ND
2	J1,J2	DK	1/4" phone jack, Switchcraft 35PM1, SC1455-ND or equiv
1	Enclosure	DK	SERPAC, 5.62x3.25x1.50 Blk, SR151-IB-ND
1	Bat Hldr	DK	Four AA battery holder w/leads, Memory Prot Devices BC4AAW, BC4AAW-ND
3	Knobs	Mouser	Eagle control knob, .55 DT18 BLK, 450-AE141
MISC.			Fellows sheet laminates
4	Screws		#4 x .25" sheet metal screws for mounting PCB in enclosure

PARTS LIST

■ **FIGURE 4. GTR_Pre-Amp Lbl4M.**

Building the Unit

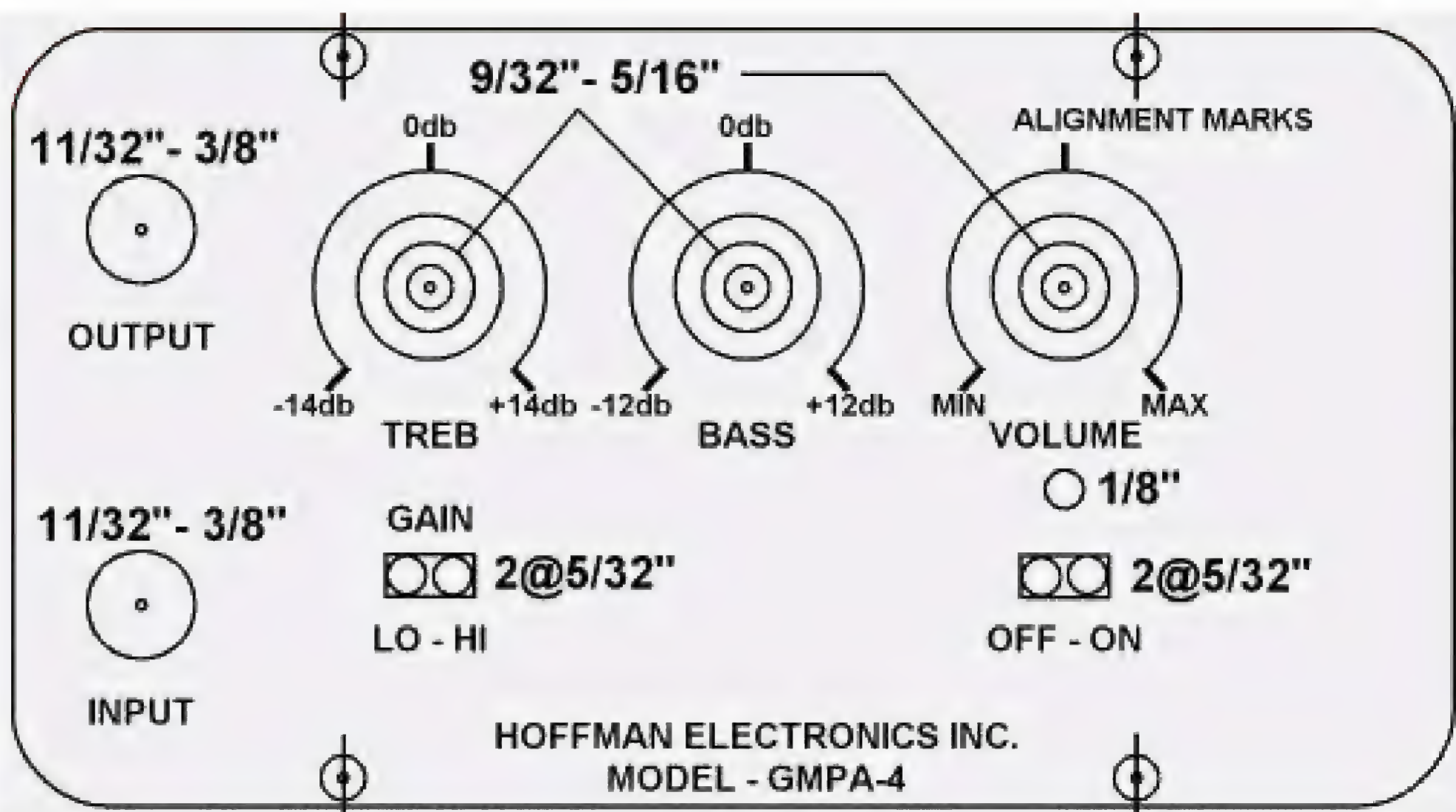
The bill of materials (BOM) includes the part numbers and sources for all of the items needed to build the unit. The Gerber files are also provided for the printed circuit board (PCB). I built my unit using a RadioShack 276-147 general-purpose PCB.

The Gerber file check print (**Figure 2**) was used to place the components and wire-wrap wire was used to connect any parts that were too far apart to use the leads to make the connection.

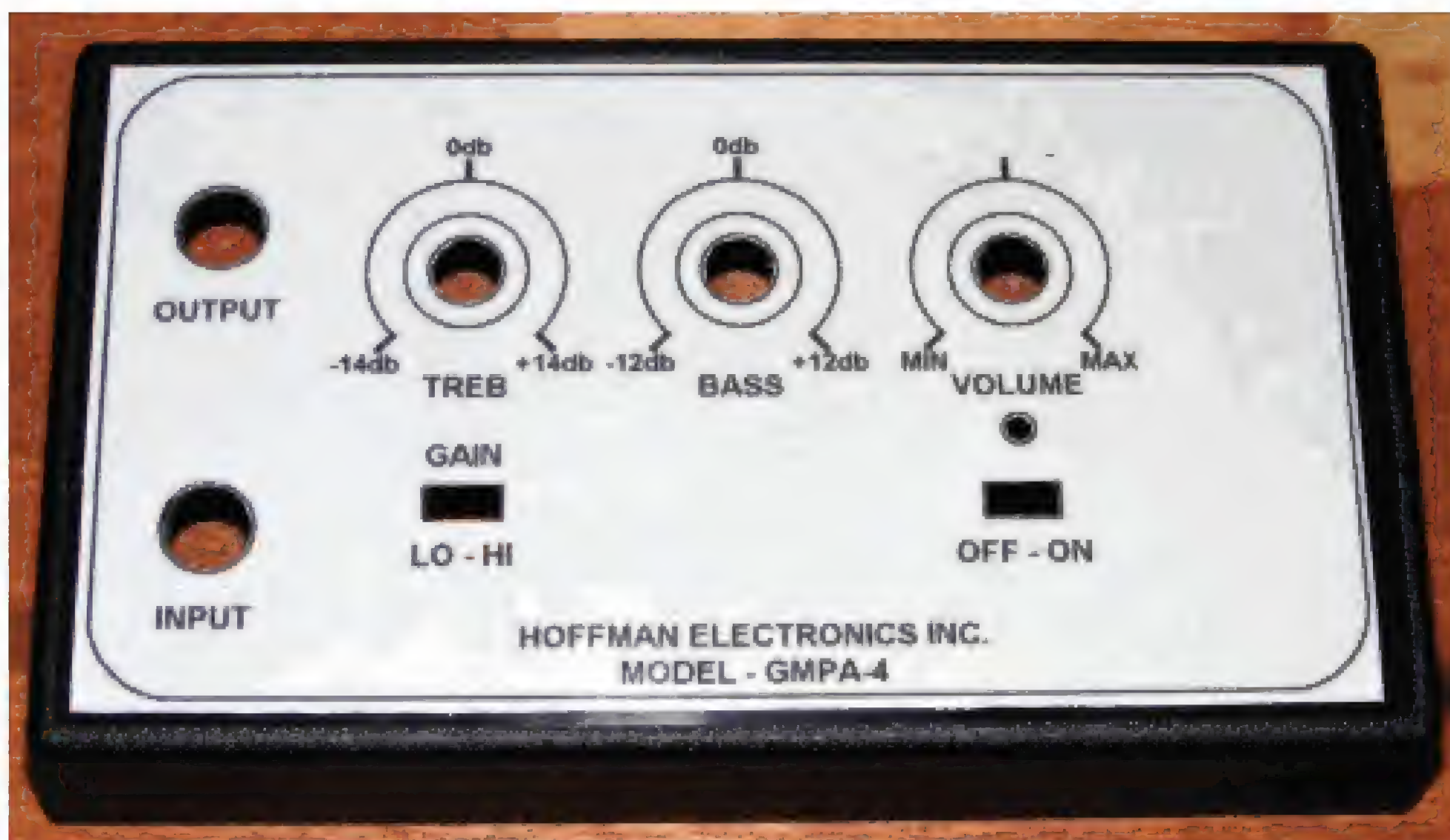
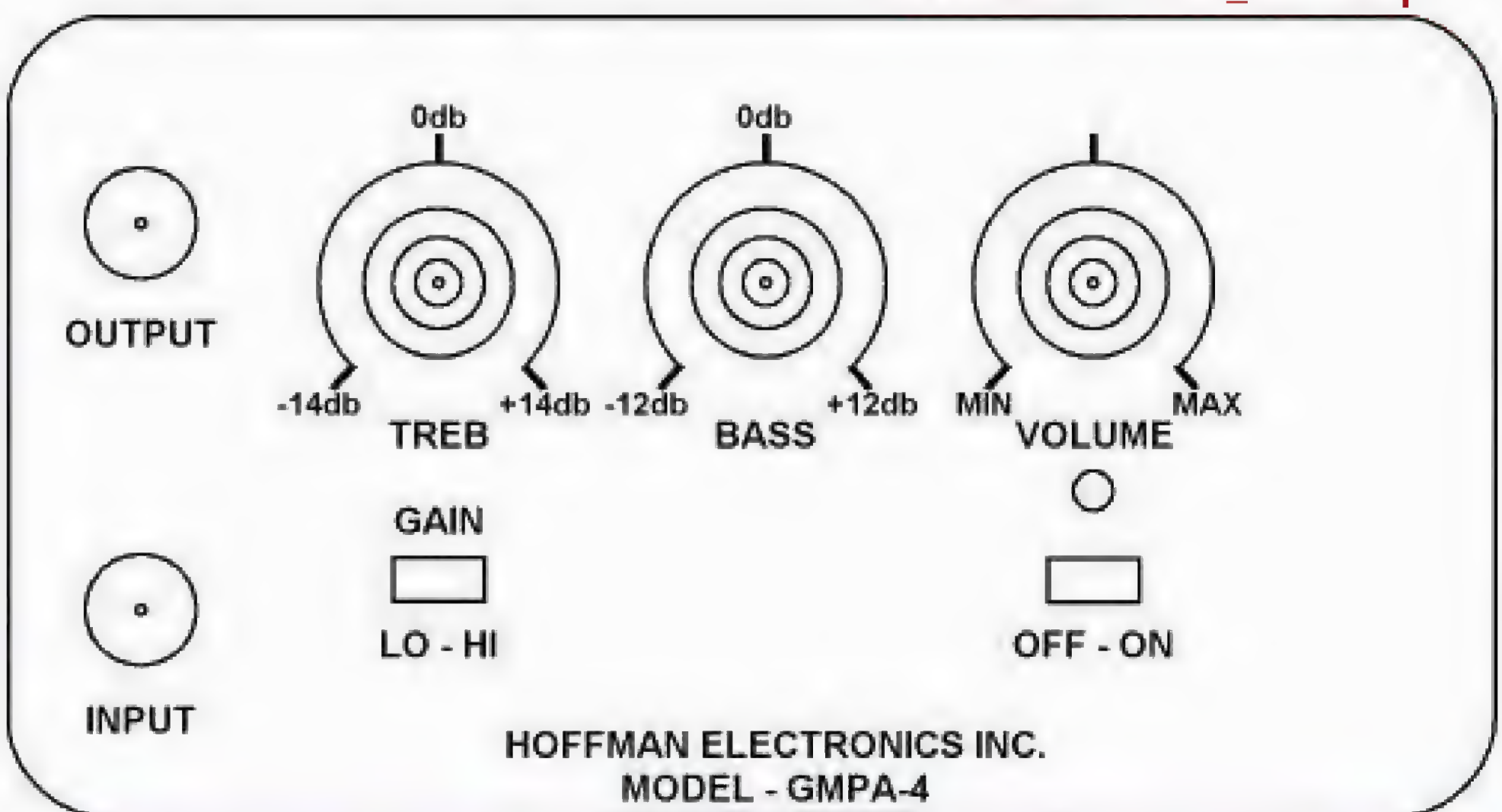
Also, since the PCB I designed was two-sided, the wire wrap allowed me to cross over wires and make connections that wouldn't have been possible on the single side of the board (see **Figure 3**).

Be careful to check all your wiring and definitely use the 14-pin socket for the MCP6024 so you can replace it if needed. Check all the ohms between pins against the schematic before applying the battery, and then check the plus input of U1D (pin 12) compared with ground. It should read about 2.2 volts.

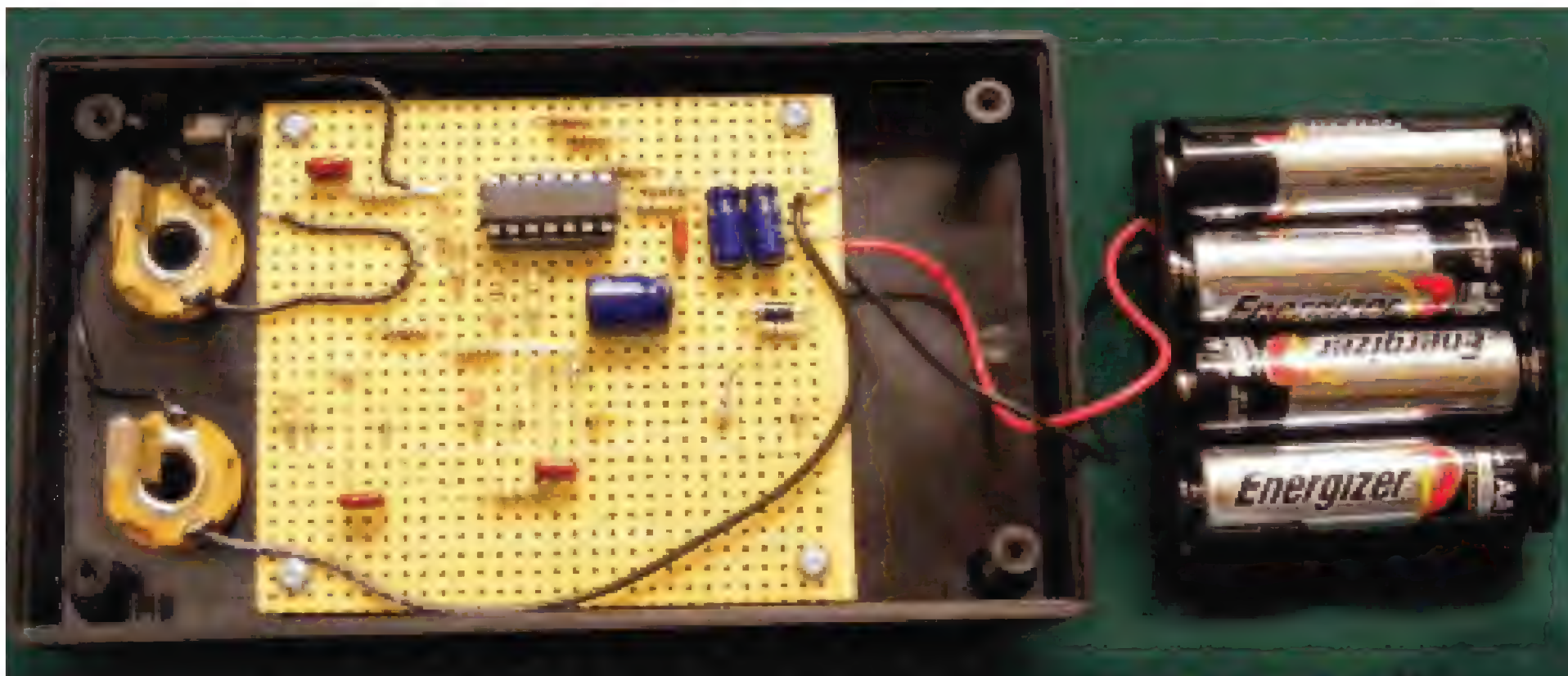
The voltage across C8 should read about 5.4 volts. If that all checks out, turn the power off and install the MCP6024 being careful to observe the pin 1 placement. If you turn it around 180 degrees, you will fry it upon applying power.



■ **FIGURE 5. GTR_Pre-Amp Lbl3.**



■ **FIGURE 6. Label applied to enclosure.**



■ **FIGURE 7. PCB installed.**

The files for the label are provided in psp, pdf, and jpg formats. They were made using Paint Shop Pro 6 at 300 dpi and measure 2.8" H x 5.1" W. I printed multiple copies of the GTR_Pre-Amp Lbl3M.jpg label and used one for the drill and filing guide on the enclosure (**Figure 4**). The alignment marks should line up with the center of the dimples in the top cover. I made pencil marks on the rim to align the label.

I also printed multiple copies of the GTR_Pre-Amp Lbl3.jpg (**Figure 5**) for the label itself. After cutting out the label, I laminated it with Fellows self-adhesive laminating sheets. After you burnish the laminating sheet to the label, cut the holes out for the phone jacks,

and mount the label to the enclosure (**Figure 6**).

Hold the phone jacks inside the enclosure to prevent them from turning while tightening the control nut and washer on the label side.

Finish the electrical assembly by connecting some 24 gauge hookup wire from the circuit board to the input and output phone jacks. Mount the board to the mounting bosses in the enclosure using small Phillips head sheet metal screws (**Figure 7**).

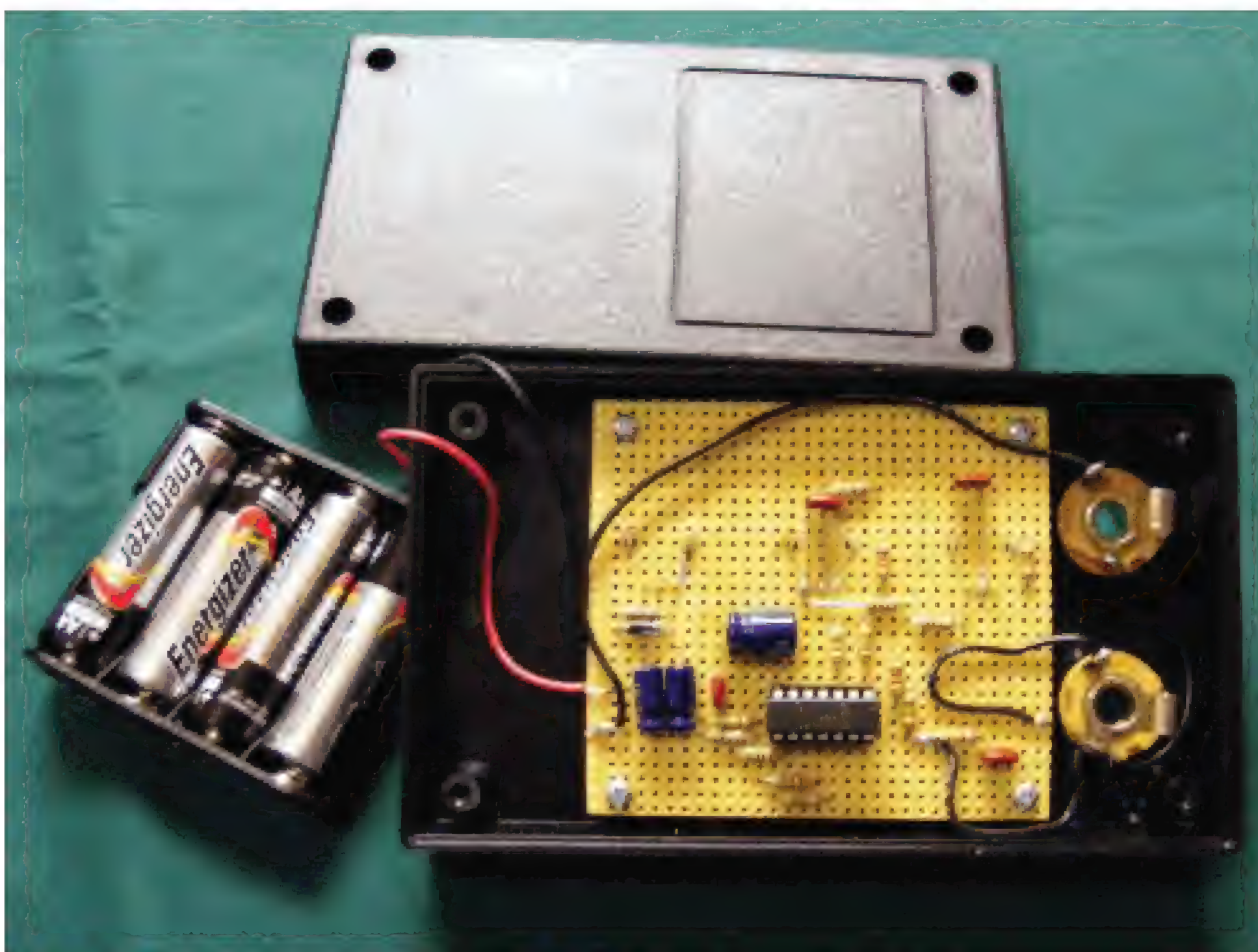
Push the knobs onto the control shafts. I used a 3/8" drill to remove two of the mounting bosses from the rear cover so that the quad AA battery pack would fit.

Finally (refer to **Figure 8**), place the battery holder over the circuit board with the batteries facing the board and secure the bottom cover to the unit with the four self-tapping sheet metal screws that are provided.

Using the GMPA-4

To use the pre-amp, plug a 1/4" shielded patch cord from your guitar into the input of the GMPA-4. Plug another patch cord from the output of the unit to the input of an amplifier, PA channel (line level or low gain), or aux input of a mixer. The pre-amp works with most guitar pick-ups, and electret or high impedance microphones.

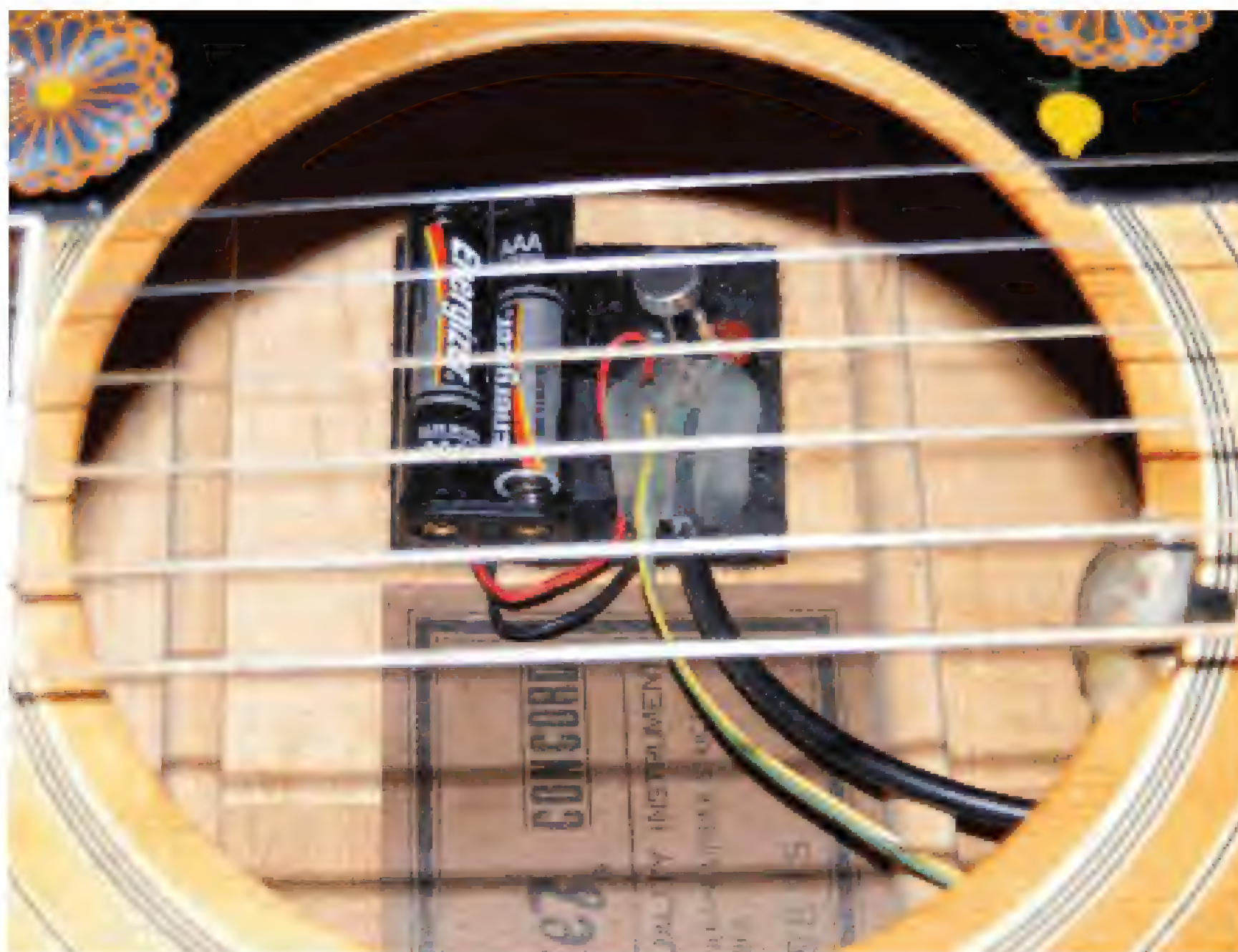
For my guitar, I have to set the treble at about +6 dB to +8 dB (3 to 4 o'clock), and the bass at about -6 dB to -8 dB (10 to 9 o'clock). This gives a very natural, balanced sound from the house PA



■ **FIGURE 8. Final assembly.**



Ron enjoying his pre-amp.



A close-up view of the Ibanez acoustic.

or amplifier for my guitar.

I adjust the volume to balance the guitar output with the stage microphone volume. It works great and no one gives me sour looks for needing to change their system channel settings. You can also plug into the aux jacks on just about any Karaoke machine, boom box, stereo amplifier, or surround sound system. Now, that is really versatile! Enjoy setting your sound the way you want it.

Final Thoughts

The thought crossed my mind about building a three- or a five-band graphic equalizer to offer more fine-tuning of the tonal profile. If readers are interested in a next generation, more advanced unit with these features, please email me and let me know. Depending on the response, I will design and build that unit. **NV**

NEW! from **velleman**

3D PRINTER KIT

Model K8200

Advanced kit requires mechanical, electronics and computer skills

Build-it-yourself 3D printer. Prints objects of maximum size of 20 x 20 x 20 cm using either PLA or ABS filament (3mm.) Extremely fast, reliable and precise even when printing at high speeds. Model K8200 is compatible with RepRap software and firmware (Free download). Sturdy aluminum construction, with heated print bed. Nine different colors of PLA filament available at \$39.00. Complete with power supply and all necessary parts. Filament sold separately.

NOW ONLY \$795.00

Part No. 32VKK8200M

(Sold as kit)

Electronix EXPRESS

TERMS: Min. \$20 + shipping. School Purchase Orders, VISA/ MC, Money Order, Prepaid. NO PERSONAL CHECKS, NO COD. NJ Residents: Add 7% Sales Tax.

In NJ: 732-381-8020
FAX: 732-381-1006

900 Hart Street • Rahway, NJ 07065
800-972-2225

<http://www.elexp.com>
email: electron@elexp.com

MakerPlot — part 2

The DIY Software Kit

By John Gavlik and
Martin Hebel

Post comments on this article and find any associated files
and/or downloads at [www.nutsvolts.com/
index.php?/magazine/article/november2013_MakerPlot](http://www.nutsvolts.com/index.php?/magazine/article/november2013_MakerPlot).

Last time, we talked about how MakerPlot can aid in your debugging efforts as they relate to your microcontroller projects. That's a given as it can really help, but since MakerPlot is a customizable software GUI it has many more applications for your micro's project than just a debug tool. That's what will be discussed in this article.

To review a bit, MakerPlot is Windows software that plots analog and digital signals that are generated by your microcontroller; this includes many of the popular microcontrollers on the market including the Arduino, PICAXE, Propeller, Raspberry Pi, etc.

The fundamental data connection between MakerPlot and your micro is the serial port; this can be a regular RS-232 serial port or a virtual comm port that's created using a USB connection through an FTDI (or similar) chip interface. That said, you can also use MakerPlot's TCP/IP link if you want to monitor your micro at a remote location. Of course, you'll need some kind of Internet or LAN connection to do this. With the proliferation of Ethernet and Wi-Fi add-on boards these days, this is easily done.

So, besides a great debugging tool with versatile data connectivity, the real power of MakerPlot is in its ability to become a virtual instrument for your micro's data. First, however, some language definitions need to be addressed.

MakerPlot lingo

Over the years, the lexicon of analog and digital electronic hardware and firmware programming terms has evolved as the names used to describe them have changed; at the same time, they still mean the same thing.

For example, a circuit board used to be called just that — a circuit board. If you're an Arduino user, you know

it as a "shield." It's fundamentally the same thing, only with a different name — mainly to distinguish it solely for the Arduino class of hardware. The same goes for firmware. Listings of source code are called "sketches" in the Arduino world. "A rose, by any other name, is still a rose," immortalized The Bard, but apparently he was never much into product differentiation. Following this reasoning, MakerPlot has a few of its own unique term definitions that we ought to get to right now to avoid any confusion later on.

In the MakerPlot world, there are three primary terminology definitions:

- **Interfaces** — The GUI screens you see on the PC monitor where the data plots and other graphics are displayed.
- **Controls** — Any of the meters, buttons, switches, or text boxes on the Interface.
- **Controller** — Any microcontroller, be it Arduino, Propeller, Atmel, Intel, PIC, etc.

That's about it. Everything else in the MakerPlot space will most likely have the same terminology and meaning that you're used to in the general electronics and software world, so that should simplify things going forward.

MakerPlot — The "Software Kit"

While we're at it, you may be wondering why we call MakerPlot a "DIY software kit."

Most microcontroller projects are hardware kits that you build and program. In that regard, think of MakerPlot as a software kit because like a hardware kit, MakerPlot gives you all the necessary pieces to construct your own Interface screens that consist of multiple types of meters, switches, buttons, labels, LEDs, plot areas, etc. That's the real power behind MakerPlot because you can do whatever you like to create the kind of Interface screens that suit your application.

We'll get into customizing MakerPlot in subsequent

articles, but for now, let's look at what comes standard with the software. Just remember that while you examine these Interfaces, in the back of your mind think of all the ways you can modify them for your own use — because you can!

Choose Your Interface

Figure 1 is what you'll see when you launch MakerPlot from the Desktop icon. This is called the Sign On Interface. You have the choice of up to 10 Interfaces that should serve over 90 percent of your data plotting needs. Just click on any of the graphics to bring one up. There are also two important Interfaces behind the

two large buttons on the left (**Figure 2**). These are the Basic “no frills” Interface and the Standard Interface — both of which are fundamental building blocks for the other GUI Interfaces — and fundamental to your eventual customization efforts.

Figure 3 shows two other large green buttons that provide you access to the *MakerPlot Guide* — the major text reference manual about all things MakerPlot. Then, there's the Arduino Code Samples button that gives you access to many useful sketches that we use to show the capabilities of MakerPlot using the Arduino series of controllers. Of course, you can modify the code to suit whatever micro you happen to be working with. So, let's take a tour of the 10 MakerPlot Interfaces.



Figure 1 . Sign On Interface.

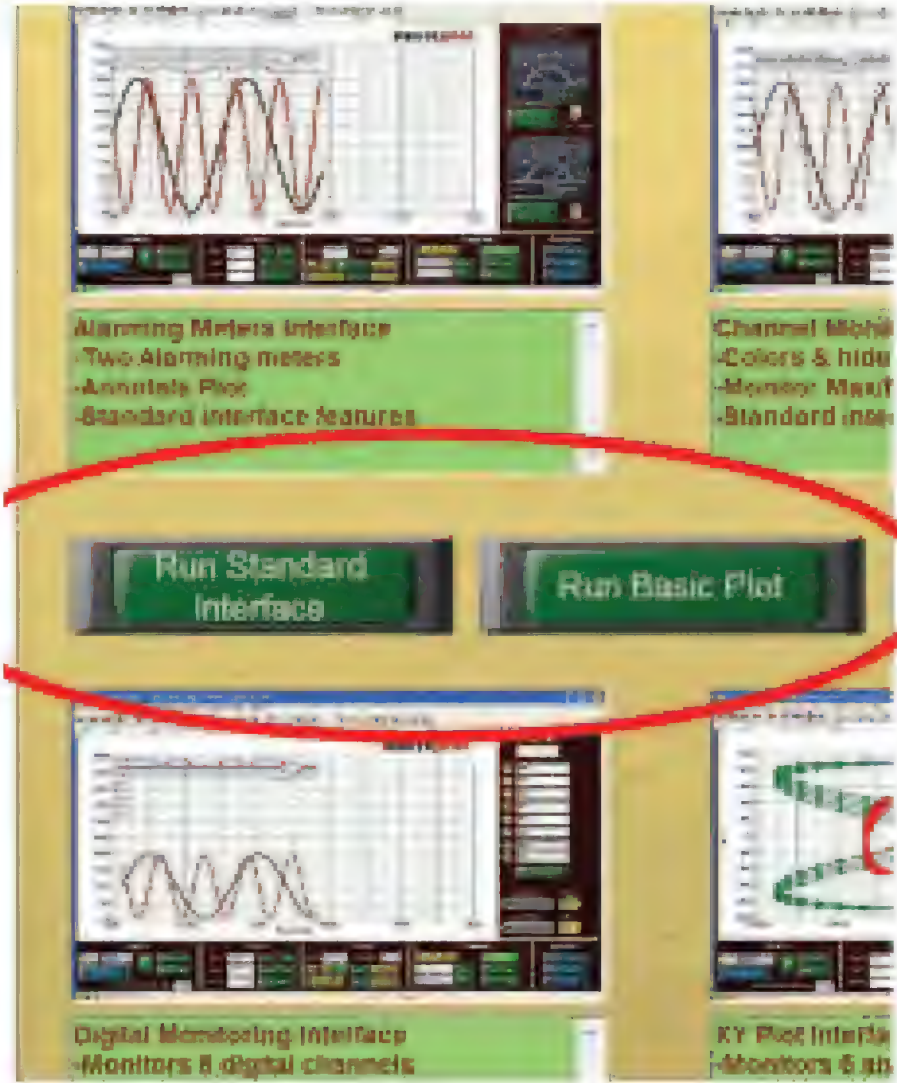


Figure 2 . Interface buttons.

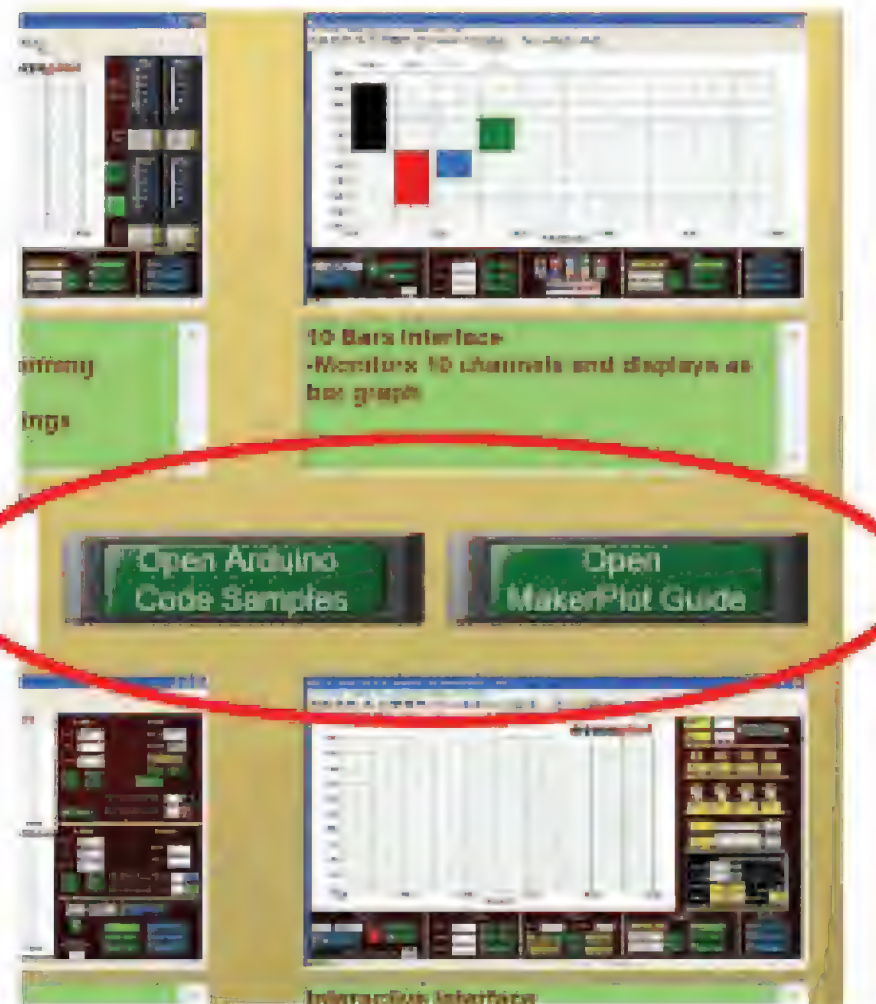


Figure 3 . Arduino sketches and MakerPlot Guide buttons.

Feedback Motion Control

The Old Way

- 1) Build robot
- 2) Guess PID coefficients
- 3) Test
 - 3a) Express disappointment
 - 3b) Search Internet, modify PID values
 - 3c) Read book, modify PID coefficients again
 - 3d) Decide performance is good enough
 - 3e) Realize it isn't
 - 3f) See if anyone just sells a giant servo
 - 3g) Express disappointment
 - 3h) Re-guess PID coefficients
 - 3i) Switch processor
 - 3j) Dust off old Differential Equations book
 - 3k) Remember why the book was so dusty
 - 3l) Calculate new, wildly different PID coefficients
 - 3m) Invent new, wildly different swear words
 - 3n) Research fuzzy logic
 - 3o) Now it is certainly not working in uncertain ways
 - 3p) Pull hair
 - 3q) Switch controller
 - 3r) Re-guess PID coefficients
 - 3s) Switch programming language
- 4) Start a new project that doesn't need feedback control
- 5) See parts in box. Feel guilty. Go back to old project
- 6) Start testing every possible combination of PID coefficients
- 7a) Apply eye drops to red, bloody, sleep-deprived eyes
- 7b) Wait, it's working!
- 7c) Decide not to do any more projects that require control systems
- 7d) Wonder why someone doesn't just make a thing that tunes itself

The Kangaroo x2 Way

- 1) Build robot
- 2) Press Autotune
- 3) Get a snack

Kangaroo x2 adds self-tuning feedback to SyRen and Sabertooth motor drivers.



\$24.99

www.dimensionengineering.com/kangaroo

DimensionEngineering

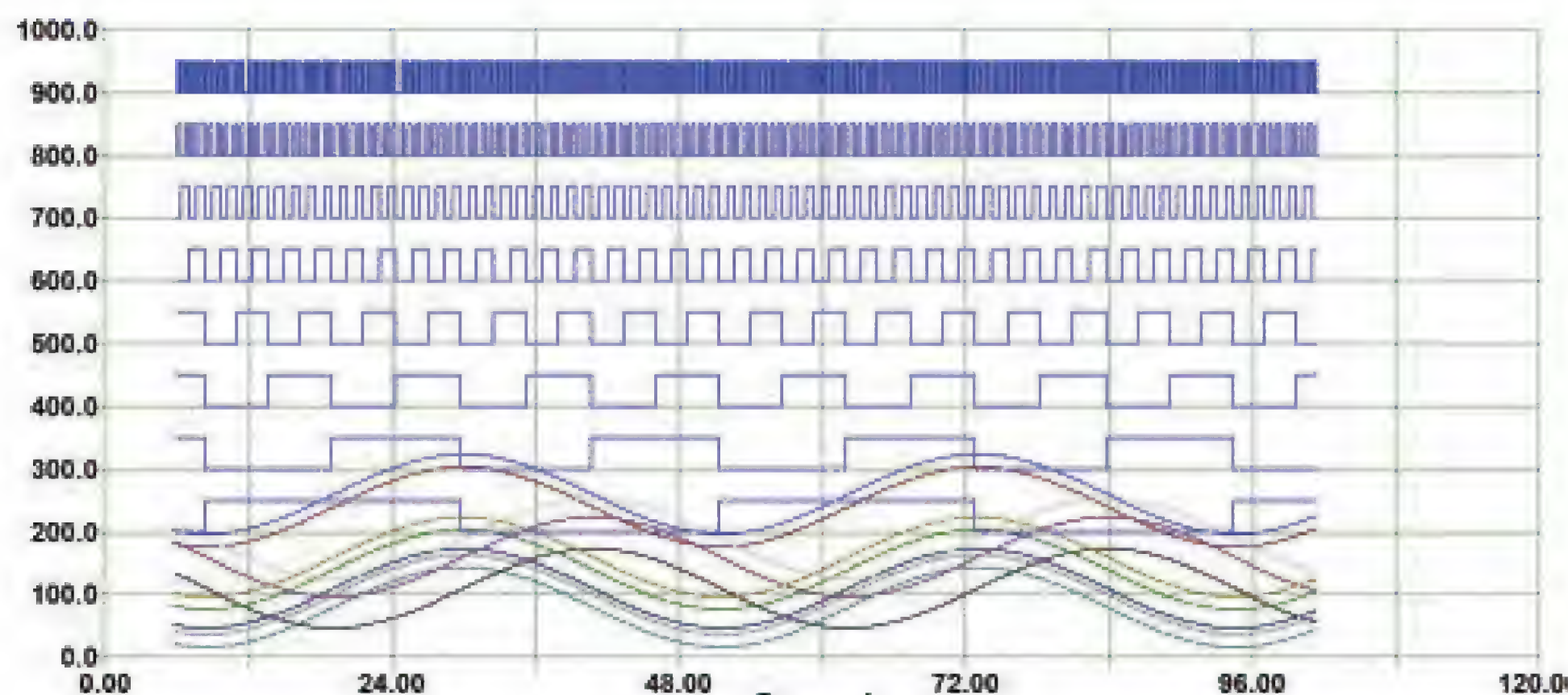


Figure 4. Basic "no frills" plot.

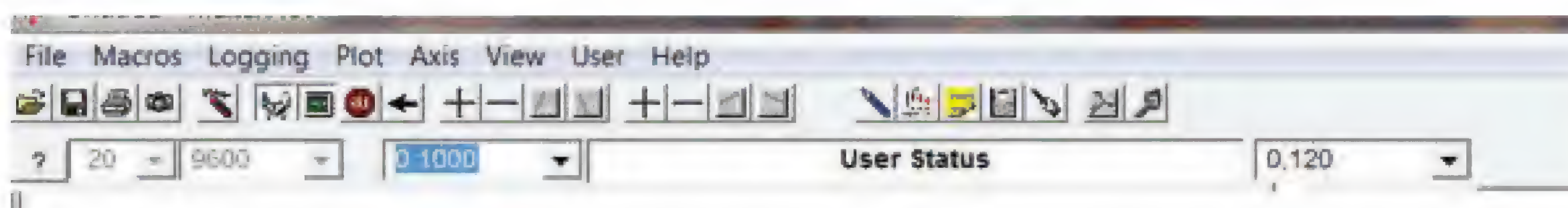


Figure 5. Basic plot pull-down menus and toolbar.

Basic "No Frills" Interface

The no frills Interface is about as "basic" as it gets — a blank plotting screen with a set of toolbar icons to set up and control your plots. If you just want to get a quick glimpse of how your micro's data is doing, this is the Interface to use. In **Figure 4**, you can see both analog and digital data being plotted without any meters or switches, etc. — just a plain vanilla data plot to give you an idea of what is being output. Like a lot of things that seem simple on the surface, there is a powerful plotting tool behind this Interface.

The toolbar icons — along with the top line menu

(**Figure 5**) — form the basis for all the other Interface's menu controls. Here, you can set up either a serial or TCP/IP connection to your micro; change horizontal and vertical scales; switch time bases from seconds, minutes, hours, and real time; scale raw A2D (analog-to-digital) output values to their correct voltage levels without any additional micro code (just the raw A2D value); save all the data to an Excel CSV formatted file; print what's on the screen; and lot's more. The Basic Interface is so extensive, in fact, that we've devoted an entire video series to it on our website. Just log on to **www.makerplot.com** and check out the Basic Plotting Video Tutorial Series. We guarantee you'll learn a lot.

Standard Interface

The Standard Interface (**Figure 6**) is where we add controls like switches, buttons, text boxes, and labels to

the Basic Interface. It still has a large plot area along with a set of menu buttons, switches, and text boxes on the bottom for most of the other Interfaces we'll discuss shortly.



Figure 6. Standard Interface.

MakerPlot is available as a FREE 30 day trial download from **www.makerplot.com**. If you like what you see and what it does, you can order it from the NV Webstore at a discounted price!



Figure 7. Control menu.

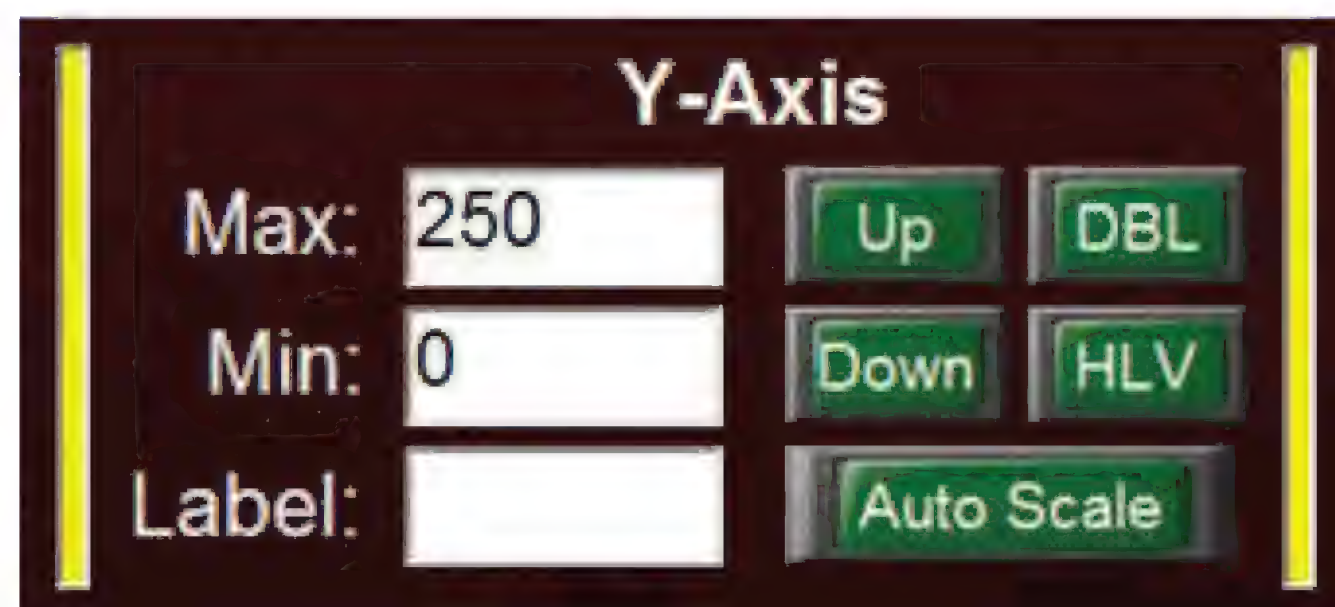


Figure 8. Y axis menu.



Figure 9. X axis menu.

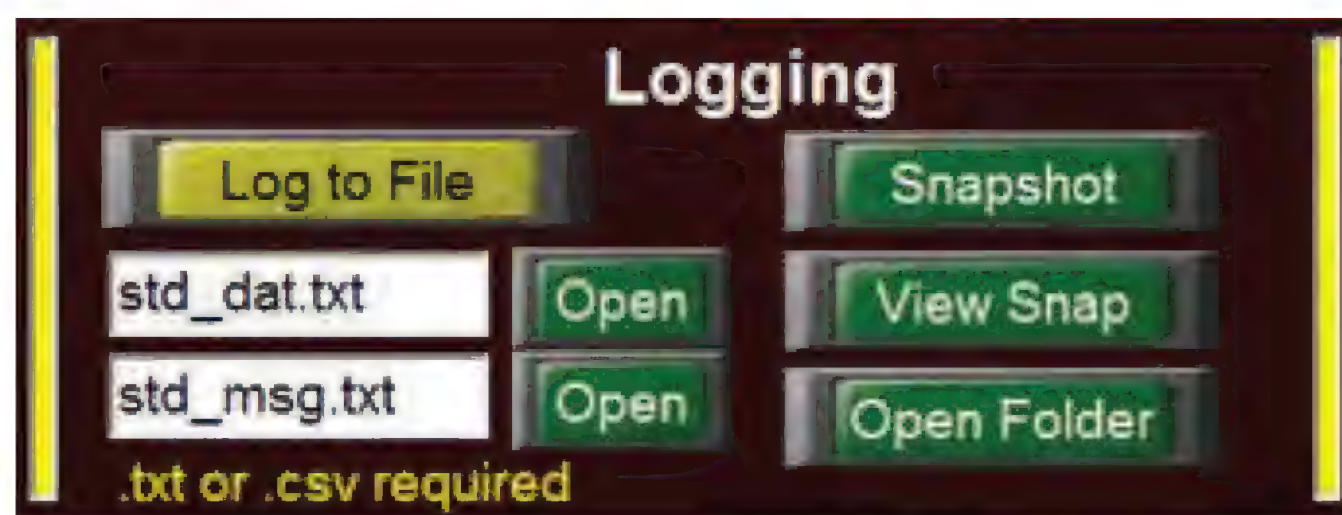


Figure 10. logging menu.

Control Menu (Figure 7)

- Select the comm port number to connect to your micro.
- Connect to your micro at the baud rate you select.
- Set the number of data points for plotting.
- Reset the plot.
- Reset the horizontal and vertical plot axis scales.

Y Axis Menu (Figure 8)

- Set the Min and Max Y scale values.
- Label the Y scale (keyboard entry).
- Shift the Y scale Up and Down.
- Double or halve the Y scale value.
- Set the Y scale amplitude to automatically adjust to your plotted data.

X Axis Menu (Figure 9)

- Set the Max time value.
- Set the plot shift percentage.
- Select Hours, Minutes, Seconds, and Real Time.
- Double or halve the time scale.
- Halt or shift the plot at Max time.

Logging Menu (Figure 10)

- Create separate log file names for data and messages.
- Select txt or csv (Excel) file extensions.
- Start and stop data logging.
- Snap a jpg image of the screen.
- View any screen capture.

Interface Menu (Figure 11)

- Save important parameters of this Interface for future use.
- Easily return to the Sign On Interface at the click of a button.
- The Standard Interface forms the basis for many of the other MakerPlot Interfaces (next).



Figure 11. Interface menu.

Alarming Meters Interface

In addition to the Standard Interface menu controls, the Alarming Meters Interface (**Figure 12**) adds two square meters to the right side of the plot area. While analog and digital data are being plotted, you can configure these meters to select one – up to as many of –

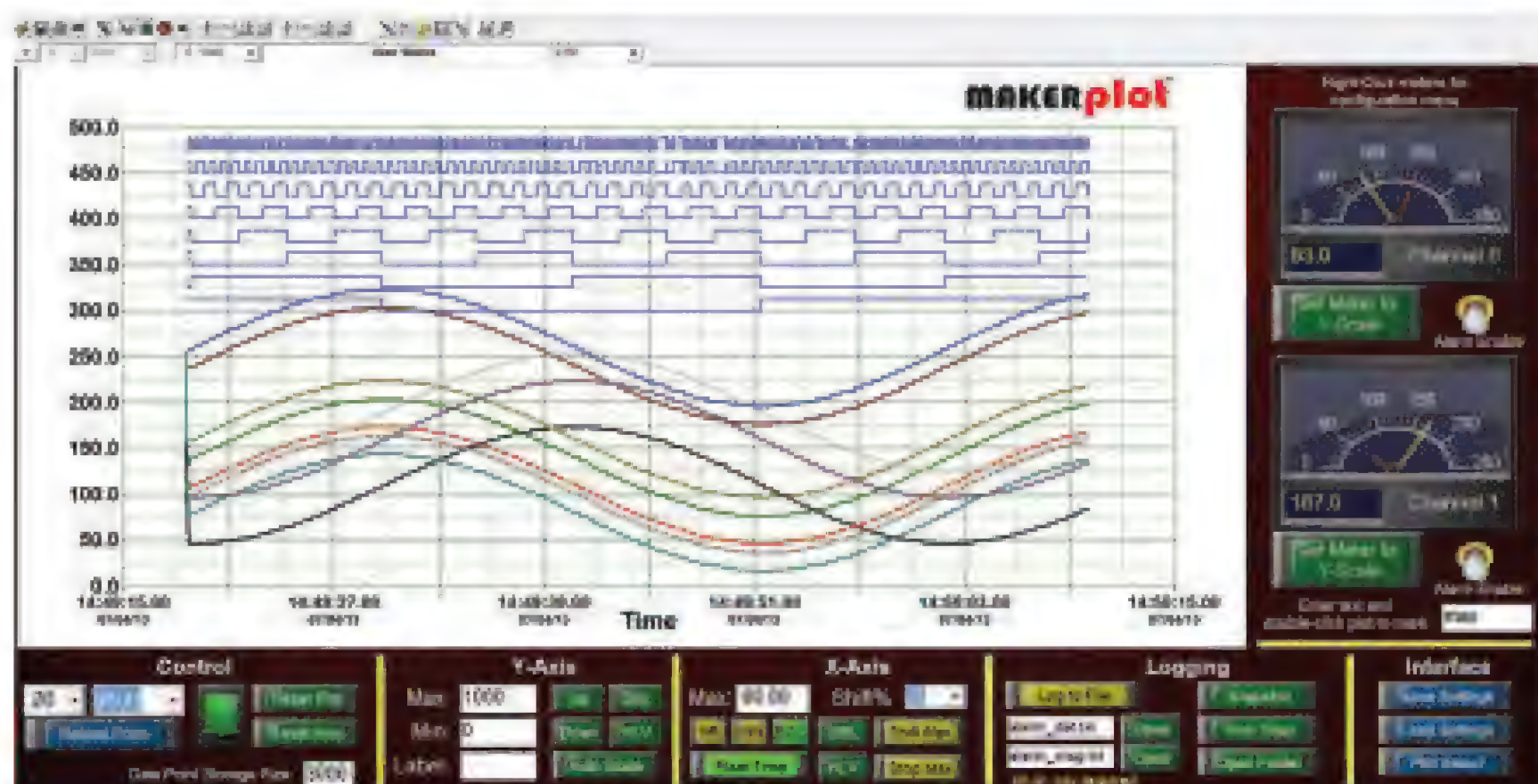


Figure 12. Alarming Meters Interface.

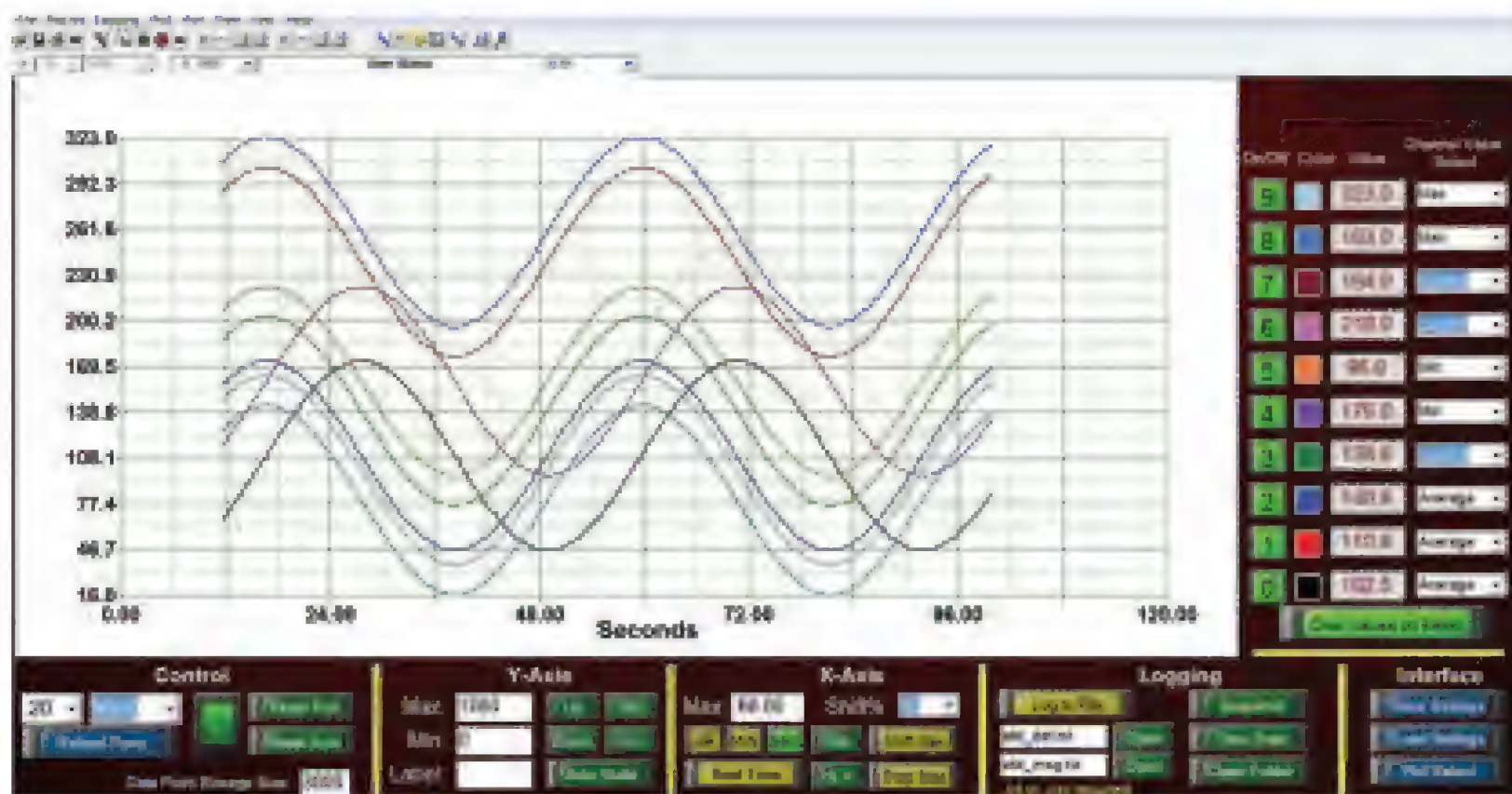


Figure 13. Channel Monitoring Interface.

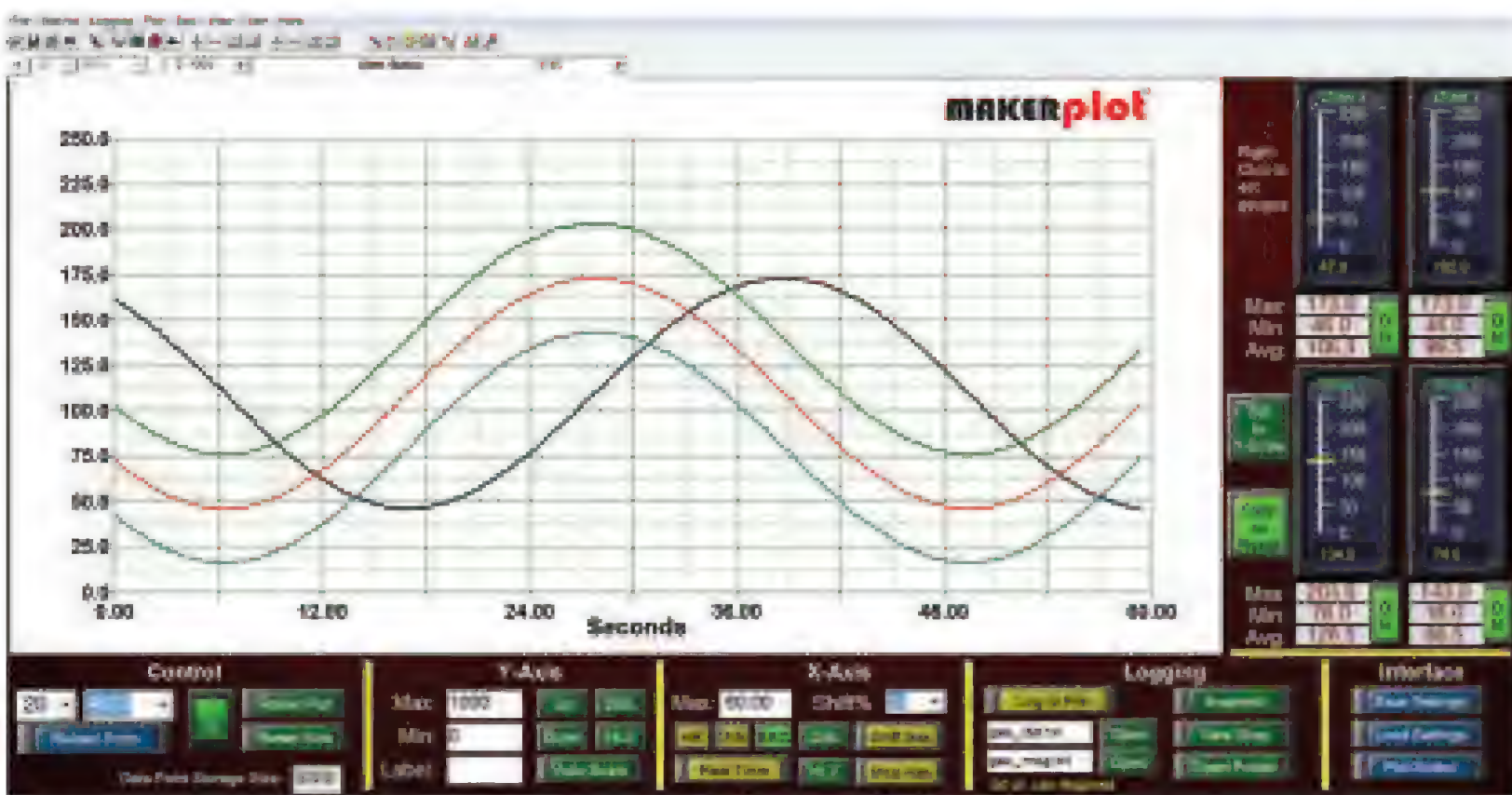


Figure 14. Four Gauges Interface.

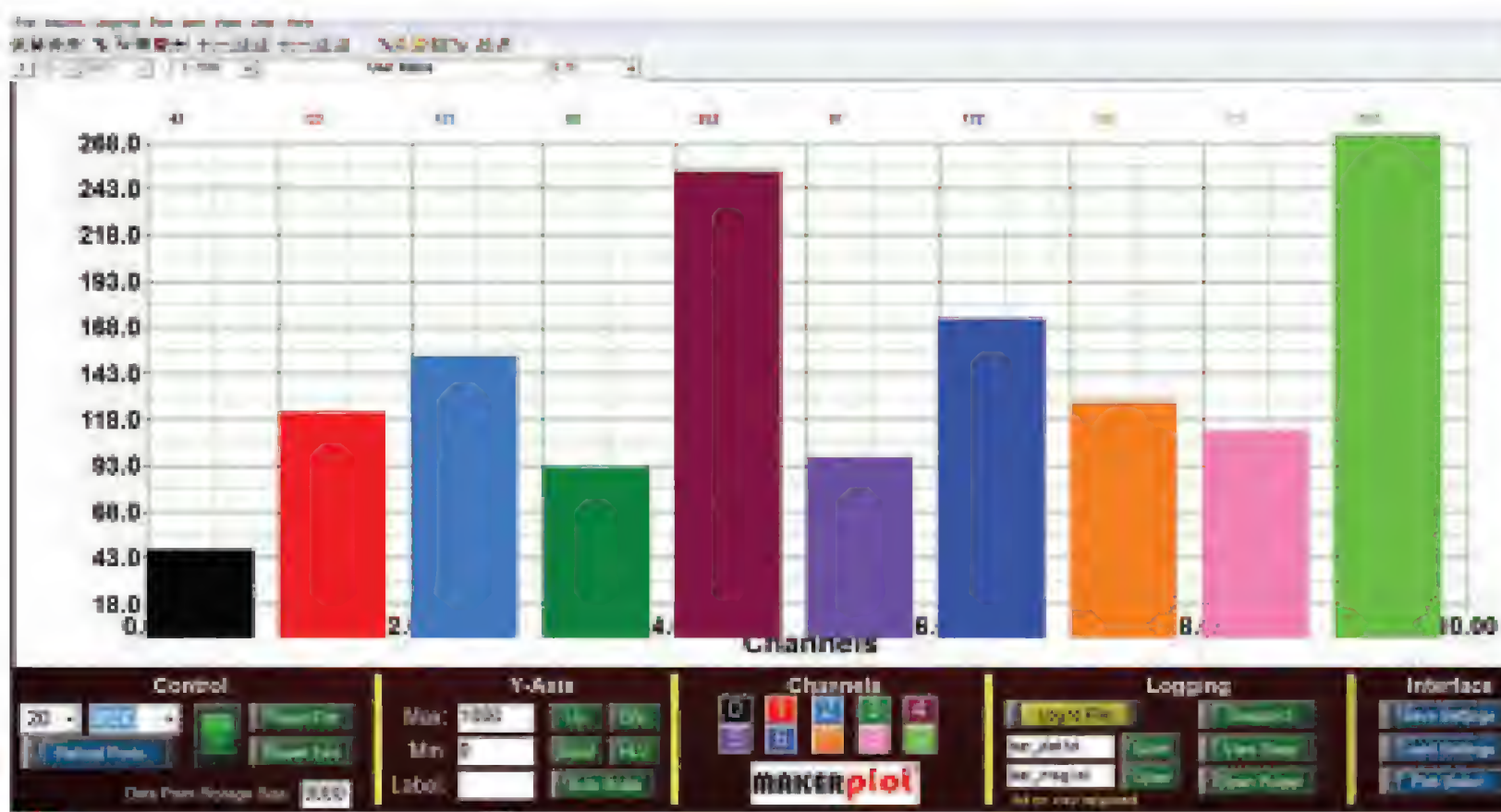


Figure 15. Ten Bars Interface.

the 10 analog channels. You can also set Min and Max alarm conditions that will alert you with audio warnings (that you can also select) when conditions are above or below limits. Finally, you can click the Set Meter to Y Scale button to automatically set the meter's full scale to the Y axis amplitude.

Channel Monitoring Interface

The Channel Monitoring Interface (**Figure 13**)

monitors up to 10 analog channels. You can do the following for each channel:

- Turn the plotted channel data On or Off in the plot area.
- Set the color of the plotted line for that channel.
- View the value of the data in numerical form.
- Configure the value as Min, Max, Average, or Current (real time).

So, if you want to monitor all your analog data at once, this is the Interface to use.

Four Gauges Interface

The Four Gauges Interface (**Figure 14**) is like the Alarming Meters Interface with two extra "vertical" meters. Here, you can monitor four analog channels at once and view the Min, Max, and Average values of each channel directly. Like the Alarming Meters Interface, you can set Min and Max alarm conditions with audio warnings, and also set the meters to the Y scale amplitude at the click of a button.

Because you can average your analog data setting, this Interface provides the Clear On Reset button that lets you reset the average analog value for each plot across the screen. Or, when not clicked, it lets you average your analog data over long time periods.

This is a great feature when you want to know what your analog data are doing for extended periods of time.

Ten Bars Interface

The Ten Bars Interface plots up to 10 analog channels in bar graph form (**Figure 15**). The amplitude of each channel is displayed at the top, and you can change the color of any bar using the menu buttons below. This is just another way that MakerPlot gives you to display analog data in another form.

Digital Monitoring Interface

With the Digital Monitoring Interface (**Figure 16**), you can plot up to eight channels of digital data and have individual LEDs flash in relation to the digital signals. You can also label each channel with text to uniquely identify it. There are two 10-segment LED indicators to monitor two channels of analog data, should you want to mix analog with digital.

X-Y Plot Interface

Instead of plotting data against time, the XY Plot Interface plots three sets of analog channel pairs (Figure 17). You can view interesting displays of data in this plotting scenario. For example, the signals that created the image shown here are all sine waves; yet, two pairs produce circles and one pair produces a line. There are lots of other interesting combinations you can generate with your data.

Dual Plot Interface

The Dual Plot Interface (Figure 18) uses two separate plot areas for plotting analog and digital data, and each plot area is independently controlled. The Dual Plot Interface is an example of how you can modify MakerPlot for your particular application. You don't have to stick with just one plot or set of menu buttons — you can have a separate plot area for each data point.

Interactive Interface

Finally, the Interactive Interface (Figure 19) is dedicated exclusively for customized, bi-directional applications that you develop for your micro. You can think of this interface as a “front panel” for your micro with switches and LEDs, along with a slider and dual joystick control. We'll get more into bi-directional control later.

Conclusion

This has been mainly a review of the off-the-shelf MakerPlot Interfaces, and it's meant to give you a brief insight into what's available to you for your plotting activity. To get a more complete look at how all the Interfaces perform, be sure to visit the Interfaces Videos menu link on the MakerPlot website as mentioned earlier.

In the next article, we'll get into how to connect your PC to MakerPlot, and then go on to show how to plot analog and digital data using the Interfaces you just read about. You'll also learn how MakerPlot can automatically scale raw A2D values to appropriate voltage levels without any coding on the micro side. This is a great feature since scaling raw A2D values does involve some degree of firmware math processing to convert from pure binary values to voltage levels.

We'll show you how your micro can output messages to MakerPlot so that you can detect when internal events happen. That's all for now so just remember: Got data? MakerPlot it! **NV**



Figure 16. Digital Monitoring Interface.

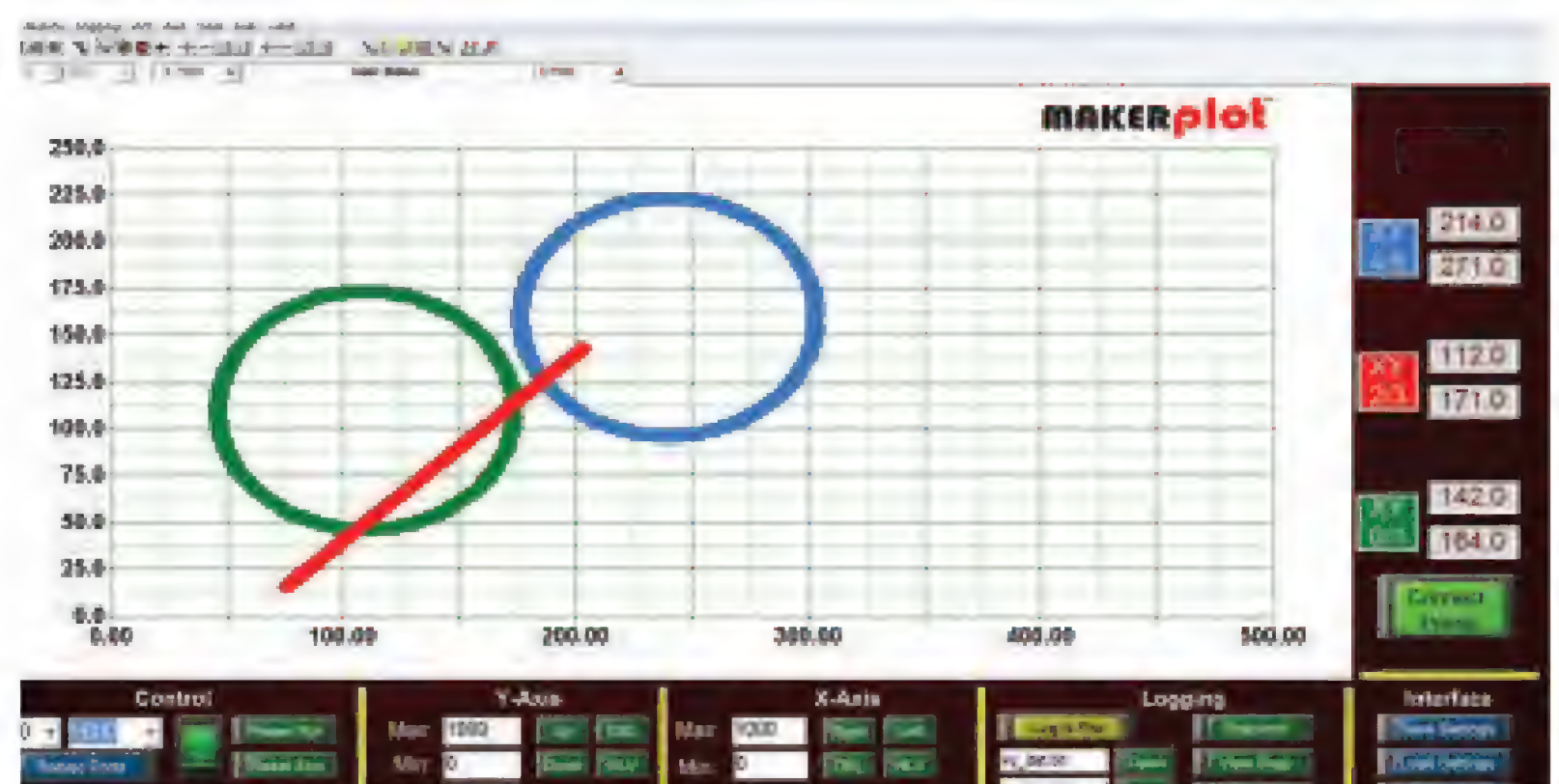


Figure 17. X-Y Plot Interface.

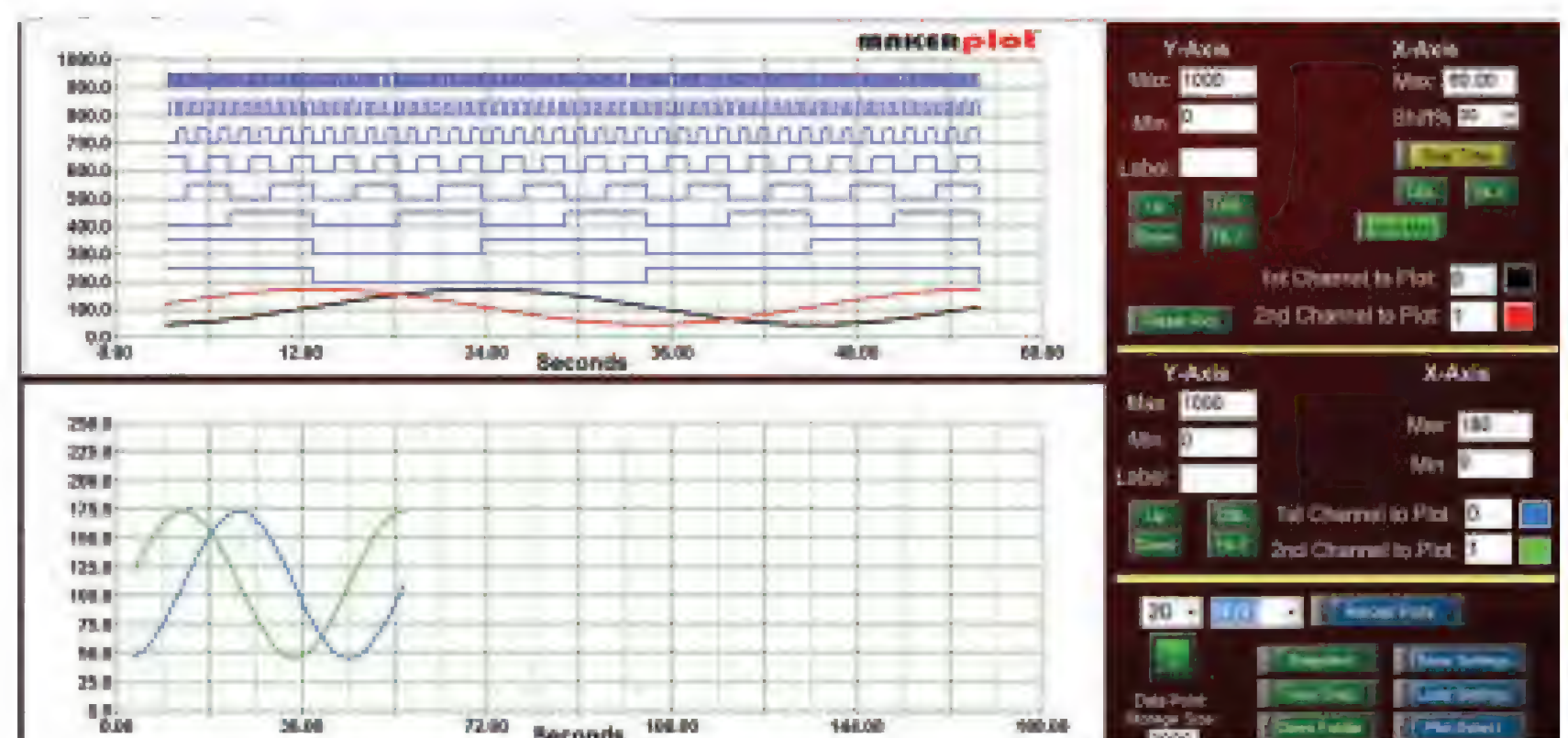


Figure 18. Dual Plot Interface.

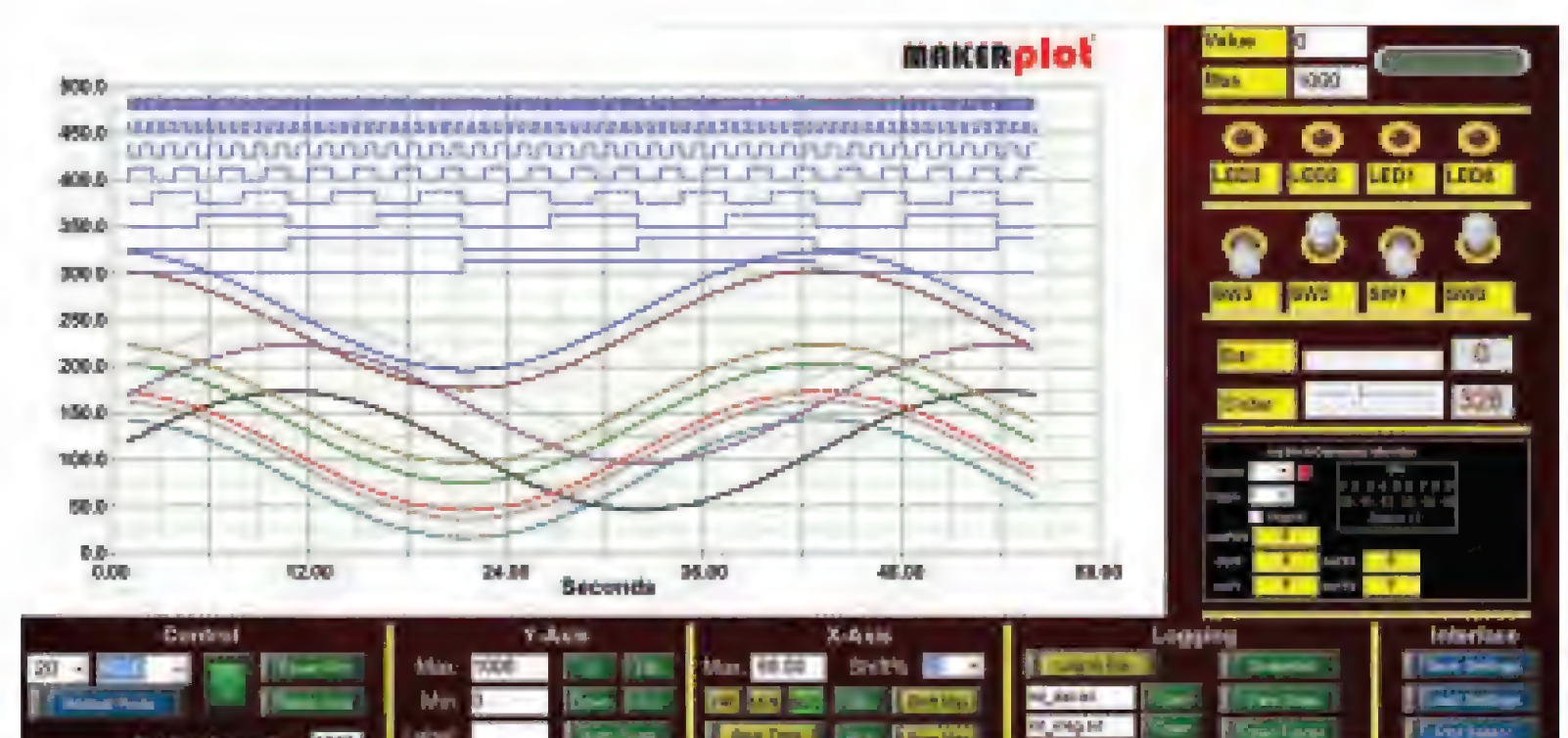


Figure 19. Interactive Interface.

Motor Controllers • Arduino • Signal Generators • FM Transmitters • Timers • Audio Amplifiers

200 KHz Function Generator

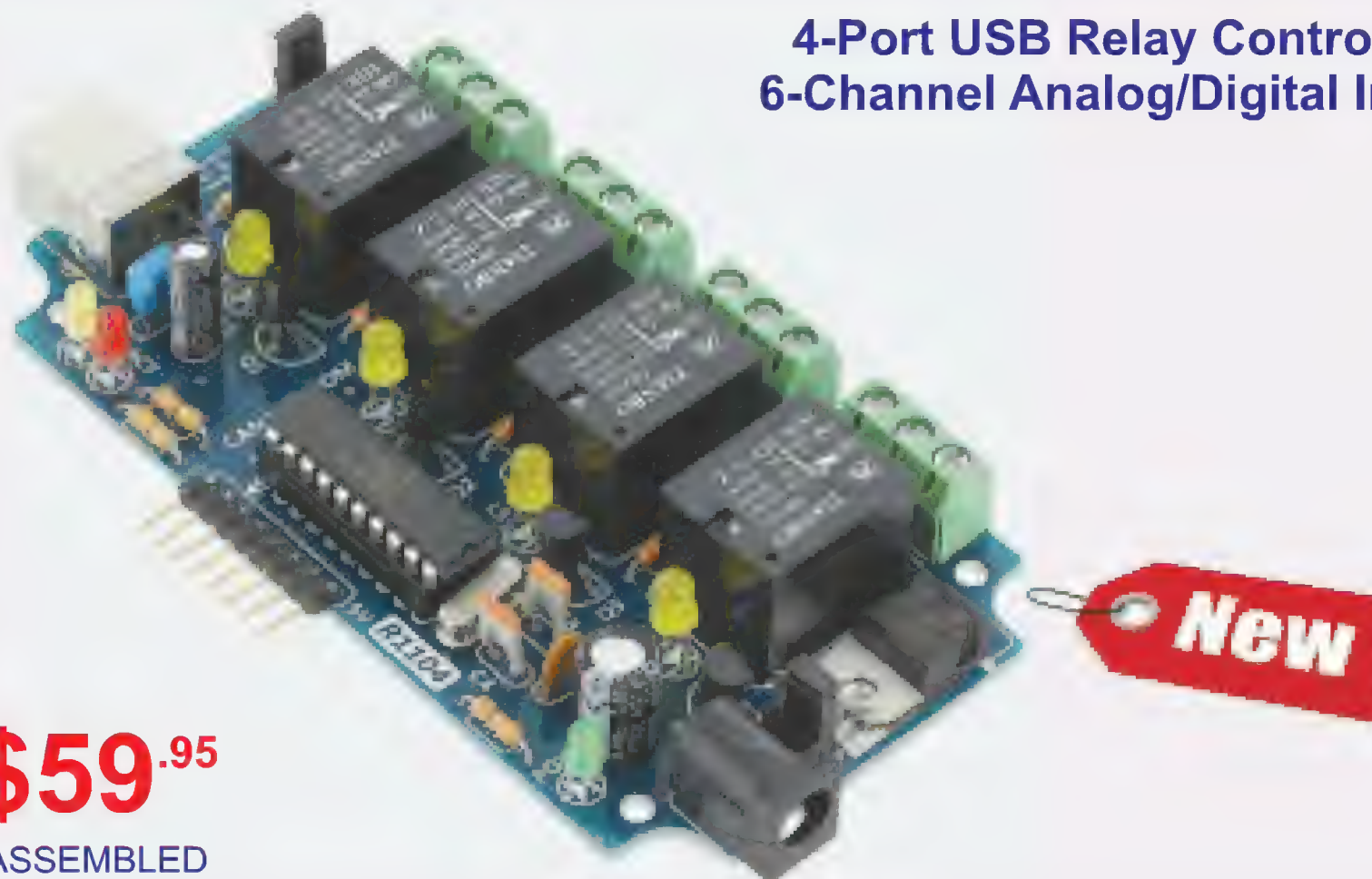


\$49^{.95}

ASSEMBLED

UK307

**4-Port USB Relay Controller with
6-Channel Analog/Digital Interface**



\$59^{.95}

ASSEMBLED

UK1104

5A Motor Speed Controller (PWM)

\$22^{.95}

ASSEMBLED



UK1102

Stereo Audio Amplifier (10W)



\$35^{.95}

ASSEMBLED

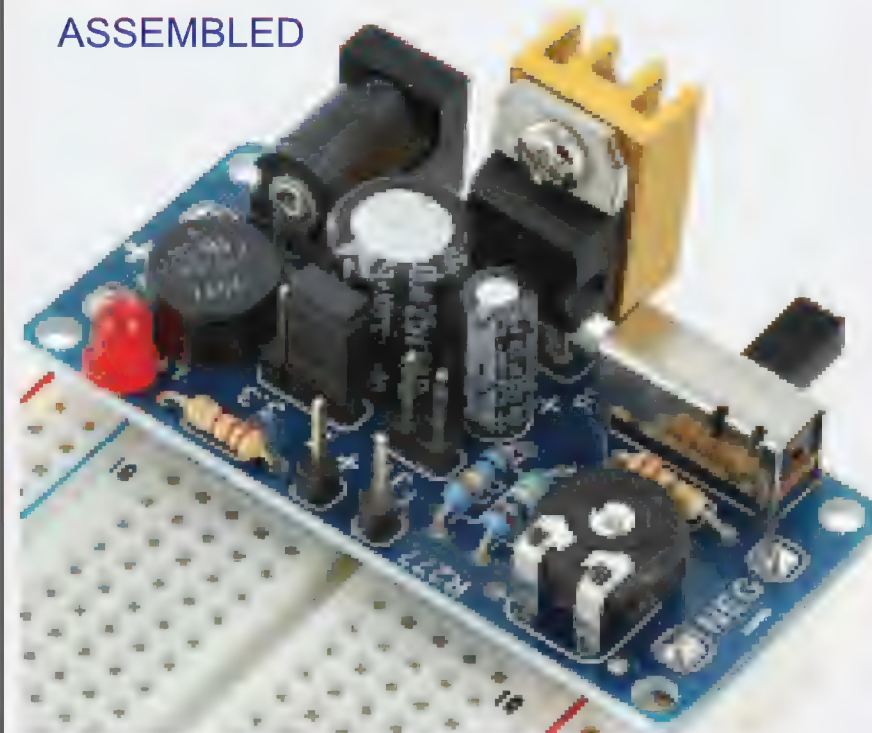
UK560

Plug-In Breadboard Power Supply

\$19^{.95}

ASSEMBLED

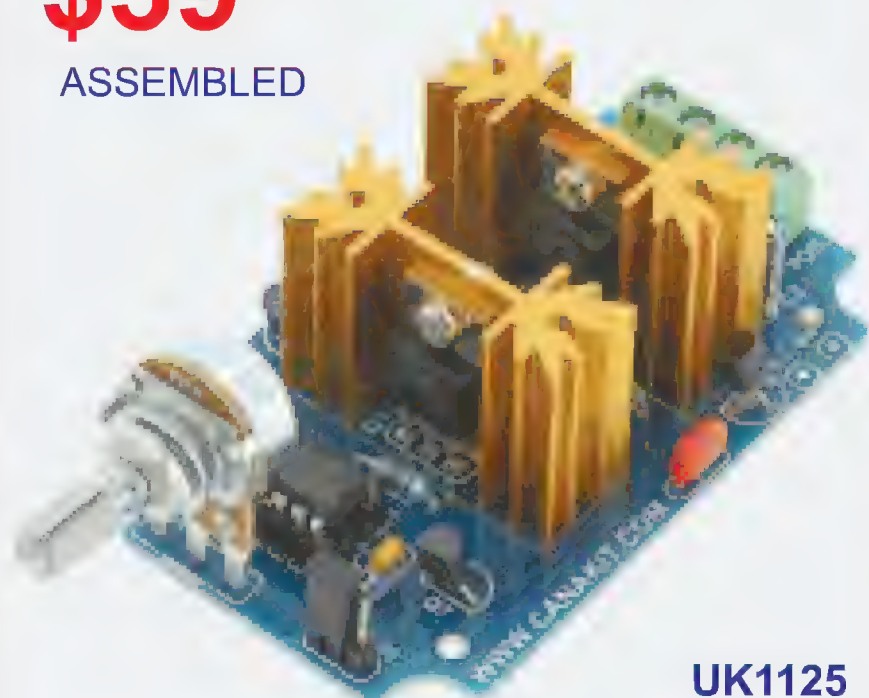
UK277



**7A Bi-Directional
Motor Speed Controller (PWM)**

\$39^{.95}

ASSEMBLED



UK1125

**USB PIC Programmer
(MPLAB Compatible)**



\$69^{.95}

ASSEMBLED

UK1300

Digital LED Timer (Up/Down)

UK258



\$46^{.95}

ASSEMBLED

Arduino Handheld Prototyper

Data Logging

Part 5

Over the past several months, we've been learning to use the Arduino handheld prototyper — a device that lets us design prototypes on a breadboard with the Arduino proto shield and then communicate with the I²C mini terminal. This is all tied together with a plastic base so that we can carry the entire development system around in our very own hand (Figure 1).



■ FIGURE 1: Arduino handheld prototyper.

Last month, we continued with the fresh air controller design and learned how to use the LCD and keys to communicate with the user. We also added a real time clock and learned how to show the date and time on the LCD. I recommend that you reread the last several articles in this series so this installment makes more sense. (Go to the article link for details on how to access Parts 1-4.)

a pen, and a bundle of hairs (yes, hairs) stretch or contract depending on humidity — also deflecting that pen. These systems are still sometimes in service in places like art museums where they have served adequately for many years to keep delicate artwork within a narrow range of temperatures and humidities. Now, of course, we use

Data Logging

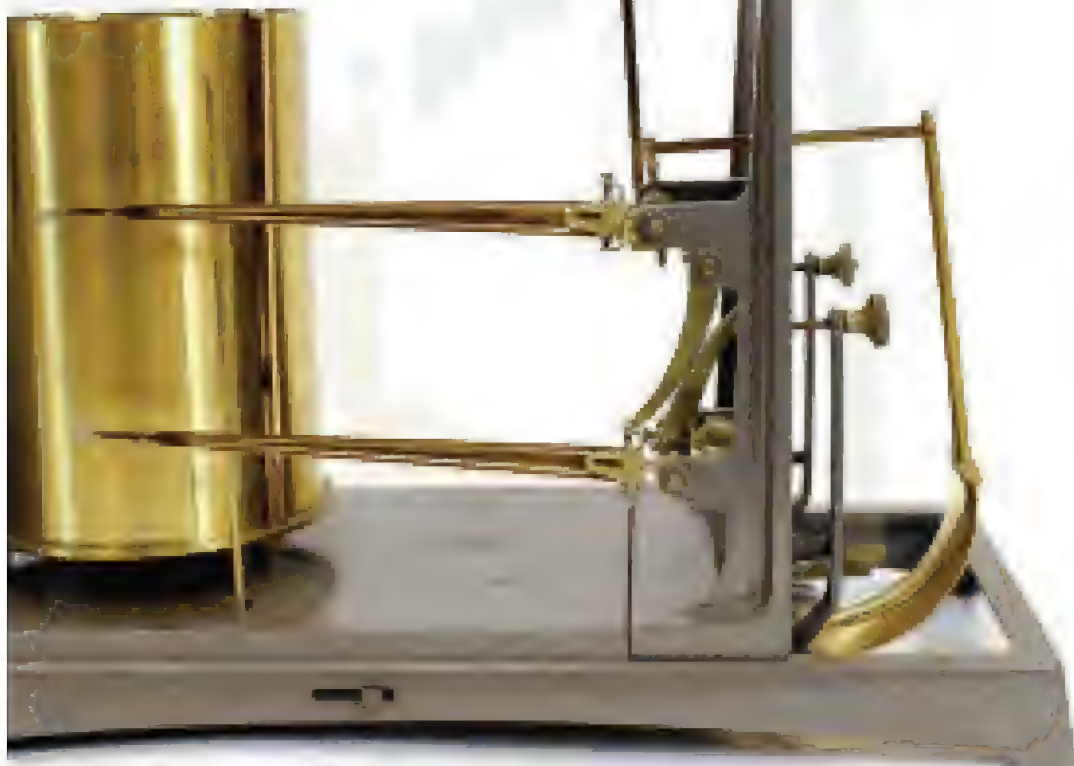
In the good old days, data loggers recorded info using a pen on a slowly moving strip of paper which produced a strip chart with X and Y parameters that depended on what was being recorded. For example, **Figures 2, 3, and 4** show a fully mechanical device that measures temperature and humidity, and records the data using pens on a chart that slowly rotates on a drum. As you can see in **Figure 3**, there is a key for winding up the drum so that it will run unattended for a week.

Temperature changes cause a bi-metal strip to deflect



■ FIGURE 2: Mechanical temperature and humidity recorder.

■ **FIGURE 3: Mechanical temperature and humidity sensing.**



■ **FIGURE 4: Mechanical temperature and humidity logging.**

electronic sensors connected to microcontrollers to do the sensing as in our fresh air controller example. We only need to add the equivalent of the pens and strip chart by recording the sensor data to computer memory. (Though you may end up pulling out a few hairs of your own.)

Recording Data With an Arduino

How much data can an Arduino hold? Arduinos now come with an ATmega328p which has 32K bytes of in-system self-programmable Flash program memory, 1K bytes of EEPROM, and 2K bytes of SRAM. Theoretically, we could use any of these to record data. We learned about computer memory — especially that on AVR — back in the June through October 2010 Workshops.

Of the three available types of memory, Flash — which is the largest — seems the most tempting to try to use to record data, and indeed it can be used as such. If your application is relatively small and you have relatively little data to store (but more than can be accommodated by the EEPROM or the SRAM), then Flash looks especially attractive. Unfortunately, while it is relatively easy to store program data at compile time [we have discussed storing preset data like strings using the features in *pgmspace.h*], it gets quite a bit more complicated to store data in Flash at run time. The bootloader does this as a matter of course, but it has to take special measures involving special timing and erasing large blocks of code and so forth that make adapting these procedures to saving data a byte at a time a bit obtuse. If there is one thing we know, it is that we are using the Arduino to avoid obtuse.

If writing data to program memory at run time was easy and a good idea, there would already be a plethora of Arduino libraries to do just that, but there aren't. So, we will only look at using the SRAM and EEPROM on the Arduino; we'll then graduate to external SD cards which do have a plethora of Arduino libraries and are a cheap and easy way to store LOTS and LOTS of data. But first, SRAM.

The SRAM

SRAM is re-writable essentially for all time and eternity. However, we are short on SRAM because it is the most expensive type of memory, and we are given the minimum deemed necessary to run the microcontroller [ATmega328p: 2K (2048 bytes)]. That said, if our data needs are small and/or we can dump the data to a PC quickly, then

we can easily create an array in SRAM for our data storage. Of course, what is 'small' and 'dump quickly?'

Let's say small means we are allocating a fourth of the SRAM which is 512 bytes, and dump quickly is to be determined. We've already discussed that one of the main attractions of our handheld prototyper is that it doesn't need to be tethered to a PC. If we are tethered to a PC, then our data memory isn't a problem. We just continuously upload the sensor data as we get it and let the PC deal with it.

Since we want to record data in isolation from a PC, let's arbitrarily demand that it only be connected to a PC once per day. That means we can record our 512 bytes in 24 hours, or 21.333... samples per hour — roughly one sample every three minutes. If we want to record the temperature and humidity, then we can do that about once every six minutes. If we want both the indoor and outdoor data, we can record it about once every 12 minutes. Is that bad? Not in my humble opinion. We should hope that the outdoor and indoor climate is changing slowly enough that 12 minutes isn't too much.

So, can we improve this? Of course. One way is to observe that neither the temperature nor the humidity will change more than a few points in a few minutes. We can again arbitrarily say that the temperature won't go up more than one degree in two minutes, and that the humidity will only change by one percent in two minutes. This means we can expect a maximum rise or drop of six degrees or percent in our 12 minute sampling interval.

Neither is true if a storm hits, so you'll have to build in some sort of exceptions which we'll get to later. Other than those rare exceptions, you should only need to record a difference of a few integers from the last reading at each new reading.

This gives us the opportunity to record temperature or humidity in a byte — capable of coding 256 values — then next only record the difference + or - in the next reading that (as we saw above) will be much smaller than the 256 a byte can hold. We can use a four-bit half-byte called a

nibble to store the change. We've seen bytes before as both hexadecimal and binary numbers such as:

Decimal: 165
Hexadecimal: 0xA5
Binary: 10100101

The nibbles are as follows:

First hexadecimal nibble: 0xA
Second hexadecimal nibble: 0x5
First binary nibble: 1010
Second binary nibble: 0101

Of course, there is no way to split the decimal value into nibbles which is one of the reasons we use hexadecimal in the first place.

Using nibbles, we can record changes between zero and 15 – the values that can be coded by four bits. We assume this will be enough to cover the changes over the previous value. However, we have also added for the complexity that the change can be positive or negative, so we'll need our nibble to be a signed data type (+ or -) that can range from -8 to 7.

For example, if we take a temperature reading that is 73 degrees and in the next sample the temperature is 70 degrees, we only need to save -3 in a nibble. Then when we extract the data, we record the 73 degrees as our first data point and we subtract the -3 to get our next data point, and so on for each subsequent data point.

Theoretically, we should only need to record the full byte once and from then on we only record the change in the following nibbles. This nearly doubles the number of samples we can log.

So, what are signed nibbles and how do we use them? The most significant bit (leftmost) in the nibble will be used for that sign: 0 for positive and 1 for negative. **Table 1** shows how each bit in each nibble translates to the signed decimal value we will use. Later, we'll write the code to use these values to derive each data point based on the original stored byte.

Now We're Talking About Nibbles?

Okay, now I'm lost. This is hardly the first time we've gotten so deep into a side issue that I've nearly lost track of what I'm doing, so let's refresh.

What we are doing is trying to store as much of our sensor data in an Arduino as is reasonably possible. To do that, we've decided that we will store an initial value for our data and then store a smaller nibble sized value for

the changes. (That process is the side issue I referred to.) It isn't really about the Arduino handheld prototyper or sensing data; it is a convenience for storing more data than we might otherwise be able to store. So, let's concentrate on fully understanding this and write a little test program for packing and unpacking nibbles. Then, we'll get back to the main thread of things.

Packing and Unpacking the Nibbles

Since we will be packing and unpacking nibbles in bytes in a byte array, we are creating a *virtual* nibble array that is twice the size of a real byte array. It's virtual because we don't actually have a nibble array; we are making up our own so that we can store and retrieve nibbles by an index number just like we would store and retrieve any other data type in a legal array. We will store the even numbered nibbles from our virtual nibble array into the high nibble of a byte in the real byte array, and we will store the next nibble – the odd one – in the low nibble of the same byte.

For example, if we want our virtual nibble array to store 0xA in the first location, 0x5 in the second, 0x4 in the third, and 0x7 in the fourth, we would pack the 0xA into the high nibble of the first byte of our byte array and then pack the 0x5 into the low nibble of that byte. We would pack the 0x4 in the high nibble of the second byte in the array and the 0x7 in the low nibble. If we had nibble sized arrays, they would look like this:

```
nibble_array[0] = 0xA;  
nibble_array[1] = 0x5;  
nibble_array[2] = 0x4;  
nibble_array[3] = 0x7;
```

These values would actually be stored in a real array as:

```
real_array[0] = 0xA5;  
real_array[1] = 0x47;
```

Now, our job is to figure out how to pack and upack the nibbles from the virtual nibble array into a real byte array. To do this efficiently, we are going to have to use some of the more arcane C operators: % (modulo), & (AND), >> (right shift), and << (left shift). Let's take a moment to review those.

The % operator finds the remainder of a division. We will pack our nibbles in the bytes by pairing two nibbles from the virtual nibble array. Each even index numbered nibble will be stored in the high nibble of a byte;

the odd indexed nibble will be stored in the low nibble of a byte.

We can use the modulo operator to find out if our index is odd or even by noting that any even number

Bits	Decimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	-8
1001	-7
1010	-6
1011	-5
1100	-4
1101	-3
1110	-2
1111	-1

Table 1. Signed nibbles.

divided by two has no remainder (equals 0), while any odd number divided by two will have a remainder. We don't care what the remainder is, only that it exists. So, we can use an *if* statement which will be true if there is a remainder and false if not:

```
if ((nibble_index % 2) == 0) {
    // number is even
}
else {
    // number is odd
}
```

Now that we can determine if the nibble index is even or odd, we will use the `&` and the `>>` and `<<` operators to pack and unpack the nibble in the correct position in the byte. Let's look at the *packNibble* function to see how this works:

```
void packNibble(uint8_t * array, uint16_t
nibble_index, uint8_t value)
{
    // the actual array position is 1/2 the
    // nibble position
    uint16_t array_index = nibble_index/2;

    uint8_t temp = array[array_index];

    if ((nibble_index % 2) == 0) {
        // number is even
        temp &= 0x0F; // clear the high nibble
        temp += (value << 4); // add to the high
                                // nibble
    }
    else {
        // number is odd
        temp &= 0xF0; // clear the low nibble
        temp += value;
    }
    array[array_index] = temp;
}
```

The *packNibble* function takes a pointer to the byte array that we use to store data in SRAM — the *nibble_index* — and the nibble value. First, we get the array index by dividing the nibble index by two. Thus, our nibble index is 256 and our array index will be 128.

Next, we get the value already stored in the array at that index value and store it in a temporary byte. Then, we decide if the nibble index is odd or even. If it is odd, we clear the high nibble in the array byte by using the `&` operator with 0x0F to mask the high nibble to 0. Then, we move the nibble value four binary positions to the left to put it in the high nibble:

```
temp &= 0x0F; // clear the high nibble
temp += (value << 4); // add to the high nibble
```

Or, if it is the even nibble, we clear the low nibble by ANDing (`&`) 0xF0 with the value to mask the low nibble to 0. Then, we add the value to the byte without needing to shift it since it is already in the correct position:

```
temp &= 0xF0; // clear the low nibble
temp += value;
```

Finally, we place the *temp* byte back in the array. Unpacking the nibble from the virtual nibble array is similar:

```
uint8_t unpackNibble(uint8_t * array, uint16_t
nibble_index)
{
    // We are storing two nibbles in each byte of
    // the array so the array index is 1/2 the
    // nibble index
    uint16_t array_index = nibble_index/2;

    // Get the array byte to unpack the nibble
    uint8_t temp = array[array_index];

    // Determine which nibble we want and return it
    if ((nibble_index % 2) == 0) {
        // number is even
        return ((temp & 0xF0) >> 4);
    }
    else {
        // number is odd
        return(temp & 0x0F);
    }
}
```

As before, we decide if the virtual nibble index is even or odd. If it is even, we AND the byte with 0xF0, shift it four positions right, and then return that value:

```
return ((temp & 0xF0) >> 4);
```

If it is odd, we just mask off the high nibble and return the value:

```
return(temp & 0x0F);
```

How simple is that? Well, not so simple really. Having a working program to demonstrate a complex principle is always a help. So, I've written a test program that demonstrates these principles (*nibble_test*) that you can download at the article link. The output to the serial monitor from running the test is shown in **Figure 5**.

Using the EEPROM

The Arduino (with the ATmega328P) has 1,024 bytes of EEPROM that was designed to store data when the device is turned off. Since we decided to dedicate 512 bytes of SRAM for our data array, we can use the EEPROM to dump our SRAM in two blocks which triples the total amount of data we can log. This gives us 1,536 bytes which (with nibble packing) can theoretically store 3,072 data samples.

I say theoretically because we do have to store some full bytes for the origin bytes. Still, this is not a lot of data from some perspectives, but it is much better than the 512 bytes we were allowing ourselves with the SRAM and no data packing. We are very fortunate to have an Arduino library for EEPROMs that we use by including *EEPROM.h* in our code. Then, we can read and write to

EEPROM using the write functions *EEPROM.read* and *EEPROM.write*. The only thing we need now is to figure out how we want to structure the three blocks of data (one in SRAM, two in EEPROM) so that we can deal with them as a cohesive unit.

What Data Do We Need to Store?

Well, obviously since we've been talking about indoor and outdoor temperature and humidity, we need to store those values. Less obvious is that for our nibble packer, we need to store the initial full values followed by the packed nibbles. We need some way to tell which are the full bytes and which are the nibble bytes.

We will also need to deal with the case where real data falls outside our nibble range. (We arbitrarily decided that our data will never vary more than -8 to +7 points between samples, but what happens when Mother Nature disagrees?) We will need to know when (the date/time) the samples were taken, as well. This tells us we need to store and retrieve several types of data, so let's look at this a little closer.

Initial Temperatures and Humidities

We will need to store the initial values for the indoor and outdoor temperature and humidity:

```
uint8_t origin_intemp
uint8_t origin_outtemp
uint8_t origin_inhum
uint8_t origin_outhum
```

Changes in the Initial Values

We have the arrays of nibbles which we must define as eight-bit bytes; later, we will write a function to pack and unpack them:

```
// We will pack our data as nibbles into 512
// byte of SRAM structured as four arrays
uint8_t inTemp[NIBBLE_ARRAY_SIZE]
uint8_t outTemp[NIBBLE_ARRAY_SIZE]
uint8_t inHum[NIBBLE_ARRAY_SIZE]
uint8_t outHum[NIBBLE_ARRAY_SIZE]
```

Dates and Times

Oh darn! We also have to record the date and time information for each and every sensor reading. That thing is 32 bits (four whole bytes or 16 nibbles), so that is going to more that eat up all that memory we just saved ... *unless we get clever again*. How about we just record the *datetime* variable once when we first start sampling. Since we will know the sampling interval, we can use the origin *datetime* and calculate the date/time when the sample was taken by multiplying the sample number times the sample interval.

If, for instance, it is sample number 37 and the interval is 12 minutes, then we add $12 \times 37 = 444$

minutes to the origin date/time. If the origin time was noon, then sample 37 was taken at 12:00 + 444 minutes (seven hours, 24 minutes) = 9:24. Wow! Now, all we need is a single 32-bit *datetime* variable and a defined sample period to keep track of all the sample dates and times:

```
// We decided to read the sensors once every 12
// minutes
#define SAMPLE_TIME 12

// knowing the SAMPLE_TIME interval we can
// calculate each
// sample datetime
datetime origin_datetime;
```

Nibble Array Size

We defined four arrays for the nibbles. Since earlier we arbitrarily decided that our data log would be 512 bytes, we must apportion our data into that size memory so that the four origin bytes and the *datetime* variable (along with the arrays) totals 512 bytes. The *datetime* is 32 bits which is four eight-bit bytes; the four initial byte-sized sensor readings give us another four bytes. That added together gives us eight bytes that we have to store at the beginning of our sampling. We have $512 - 8 = 504$ bytes left for our nibbles; 504 divided by four nibble arrays gives us 126 bytes per array, so we define:

```
#define NIBBLE_ARRAY_SIZE 126
```

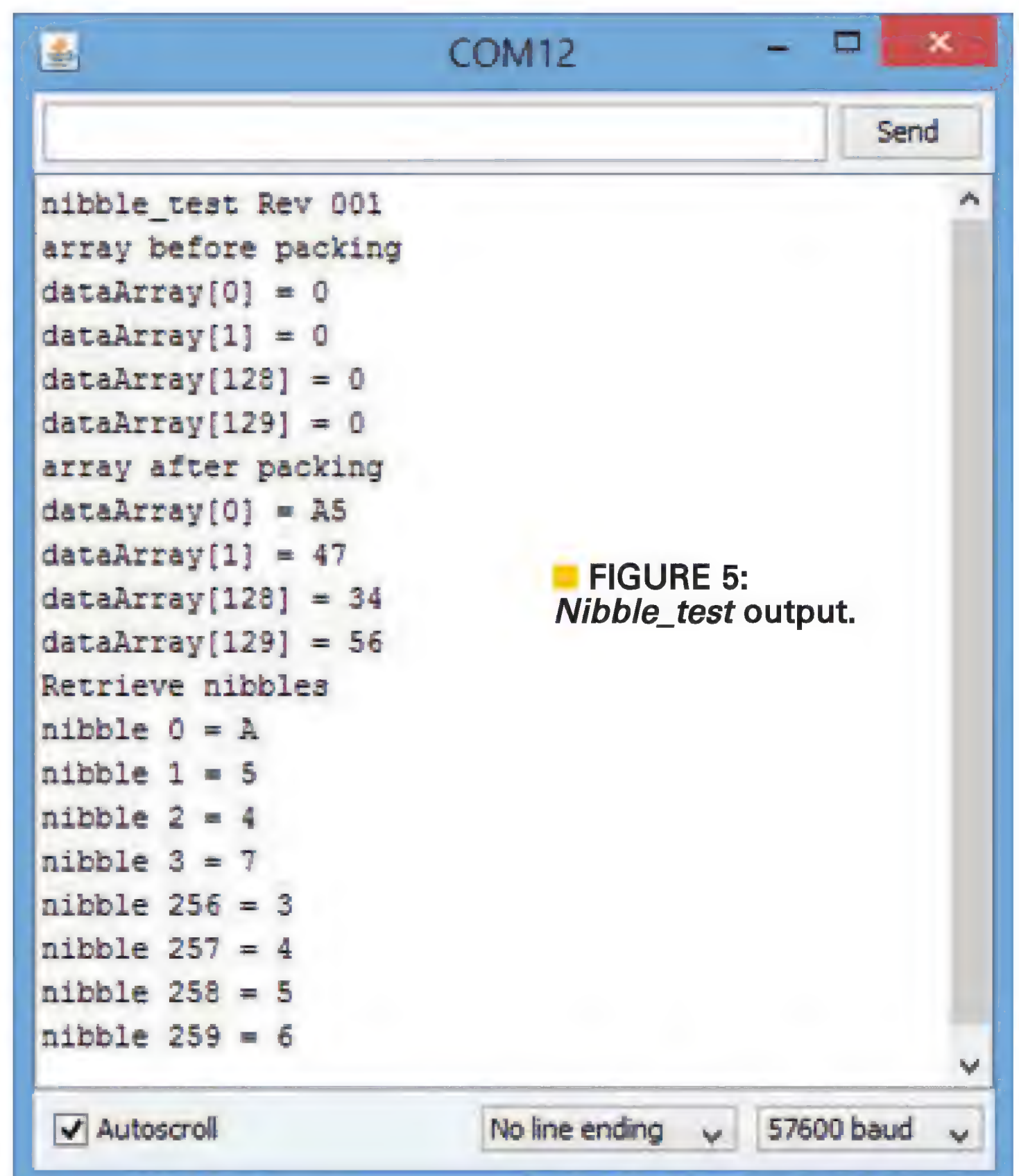


FIGURE 5:
Nibble_test output.

Remember that the 126 bytes will store twice that — 252 nibbles.

Let's Use a Data Structure

As discussed previously, we will fill up our SRAM and then transfer the values to EEPROM, giving us three separate sets of data. Since each of these sets is structured the same, we can benefit from defining a single data structure. We will first fill the SRAM instance of this

structure and then when it is full, we will copy the raw bytes from it into the first 512 bytes of EEPROM.

We will then clear the SRAM memory for the structure and begin our second set of sampling data. When the SRAM fills up again, we will copy the data to the second 512 bytes of EEPROM, and again clear the SRAM and start sampling over. We will use three instances of the following data structure:

```
struct fac_log {
    datetime origin_datetime
    uint8_t origin_intemp
    uint8_t origin_outtemp
    uint8_t origin_inhum
    uint8_t origin_outhum
    uint8_t inTemp[NIBBLE_ARRAY_SIZE]
    uint8_t outTemp[NIBBLE_ARRAY_SIZE]
    uint8_t inHum[NIBBLE_ARRAY_SIZE]
    uint8_t outHum[NIBBLE_ARRAY_SIZE]
}
```

If the SRAM fills up a third time before we download the data to the PC ... we didn't plan well, did we? We could figure out a way to trash the oldest data and keep only the most recent stuff until somebody downloads it, but we are already getting too complex. So, let's just stop sampling data and leave it up to the user to figure this part out.

Then a Storm Hits ...

So, now we've got a handle on the data during normal times, but what about that exceptional case of a storm passing through and making a change outside of our -8 to +7 range? Again, we'll have to make an arbitrary assumption and just declare that if any value falls outside our range, we will declare an exception by setting all four nibbles at that sample time to -8.

Let's assume that we will never see all our sensor values drop exactly eight points from one interval to the next under normal conditions, and we'll use this 'fact' to create an exception state that the software can see in the stored data and then compensate for it.

Even if the rare case happens that all four values do — in fact — drop exactly eight points, we only lose a little data space since the solution to the exception involves storing four new initial bytes. So, really, all we've

ALL ELECTRONICS

www.allelectronics.com

Order Toll Free 1-800-826-5432

15 AMP GROUNDED OUTLET

Black 3-conductor grounded outlet. Snap-mounts into 0.9" square chassis hole. 0.187" QC/ solder terminals. Rated 15A/125Vac. UL, CSA.



CAT# ACS-58

\$1.00
each

10 for 90¢ each
100 for 80¢ each

12VDC SPDT 40A AUTOMOTIVE RELAY

12 Vdc, 88 ohm coil. Contacts rated 40A. Plastic flange for bulkhead mounting. Mounts in standard automotive relay socket.



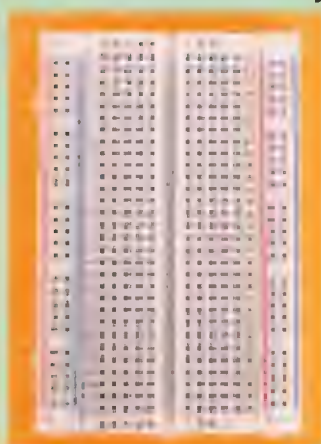
CAT# RLY-351

\$2.55
each

10 for \$2.25 each

SOLDERLESS BREADBOARD, 400 CONTACTS

Accommodates all sizes of dips and discrete components. Interconnect with solid hook-up wire. Wires and components can be re-used many times without damage to board or components.



CAT# PB-400

\$4.00
each

DPDT ON-ON MINI TOGGLE SWITCH

RATED: 6 amp @ 120 Vac. Solder lugs. 0.43" long handle.



CAT# MTS-8

\$1.75
each

10 for \$1.60 each
100 for \$1.35 each

DOOR LOCK ACTUATOR

12 Vdc motor-drive actuator for automotive door locks. Nylon plunger moves 0.75". Push or pull, depending on polarity. Overall dimensions (retracted): 5.50" x 2.45" x 1.25". Rubber boot protects against moisture and dust.



CAT# DLA-1

\$5.50
each

48 VDC 0.38A POWER SUPPLY

Phihong #PSC18U-480. 5' cord w/ molded 2.5mm coax power plug. Center positive. cULus, CE, RoHS.



CAT# PS-480

10 for \$7.00 each

\$7.50
each

SNAP-ON FERRITE BEAD

7mm (0.27") I.D. snap-bead. 32.5 x 19.0 x 19.5mm case.

CAT# FB-103

\$1.50
each

10 for \$1.35 each
100 for \$1.15 each



COLOR HEAT SHRINK TUBING ASSORTMENT, 154 PCS

4 inch lengths of 6 different diameters:

1/16" - 56 pcs
3/32" - 35 pcs
1/8" - 28 pcs
3/16" - 18 pcs
1/4" - 9 pcs
3/8" - 8 pcs



CAT# HS-4900

\$11.95
each

done is unnecessarily reset our starting values. However, the data will still be correct for the subsequent readings.

Logging the Exception

Under the condition of any sensor reading byte being outside the nibble range of -8 to +7, we will record the four nibbles (two bytes) as 0x88, 0x88. We will follow these two bytes with four new full byte initial values for each of the indoor and outdoor temperature and humidity readings. After this reset, we will resume packing the data in nibbles.

Reading the Exception

When the software that reads the downloaded sensor data sees the two bytes for a single sensor reading, it will respond to those values as an exception, discard the values, and then read the next four bytes into the initial sensor reading. It will use them as the actual sensor reading for that time period. We'll assume each subsequent reading will be in nibble format and used to calculate the true reading from the prior value.

These unpredictable exceptions mean that we won't actually know how much data we can store until we've finished since the exceptions will take up extra space. We

can hope not to see many exceptions and if we do, well ... we can always write more code.

Again – Too Much!

Exceptions? I'm already confused by all the different kinds of data we are using! We've got an origin date/time, an origin indoor temperature, an origin outdoor temperature, an origin indoor humidity, an origin outdoor humidity, and then nibbles for the changes in indoor and outdoor temperatures and humidities. Now, we also have a special exception state that resets stuff ... my head just sprung a leak.

I'm going to stop here and write a test program so that I can see all this in action before attempting to hook it up to real world sensors and actually log the data. Of course, we are out of space again, so you can find this code (*fac_data_logger_test*) at the article link.

Next month, we'll hook this up to the sensors and vastly extend the amount of data we can log by using an SD card. This will turn our modest handheld prototyper into a real beast capable of logging gigabytes of data! **NV**



AP CIRCUITS
PCB Fabrication Since 1984

As low as...

\$9.95
each!

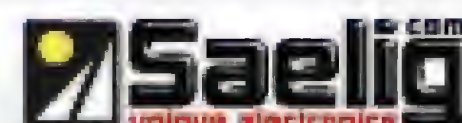
Two Boards
Two Layers
Two Masks
One Legend

Unmasked boards ship next day!

www.apcircuits.com



SCOPE DEALS



WORLD'S SMALLEST

World's smallest MSO! This DIP-sized, 200kHz, 2-ch scope includes a spectrum analyzer and Arbitrary Waveform Generator. Measures only 1 x 1.6 inches in size! - XPROTOLAB **\$49**



6-IN-1 SCOPE

2-ch 2MSa/s 200kHz 11-bit scope, 2-ch spectrum analyzer, 2MSa/s 8-bit AWG, function generator, network analyzer, and PWM/Digital I/O. - CGM-101 **\$99.95**



30MHz SCOPE

Remarkable 30MHz, 2-ch 250MS/s sample rate oscilloscope. 8-in color TFT-LCD and AutoScale function. Includes FREE carry case + 3 yr warranty! - SDS5032E **\$299**



60MHz SCOPE

Best selling 60MHz 2-ch scope with 500MSa/s rate + huge 10MSa memory! 8" color TFT-LCD. Includes FREE carry case! - SDS6062 **\$349**



100MHz SCOPE

High-end 100MHz 2-ch 1GSa/s benchscope with 1MSa memory and USB port + FREE scope carry case. New super low price! - DS1102E **\$399**

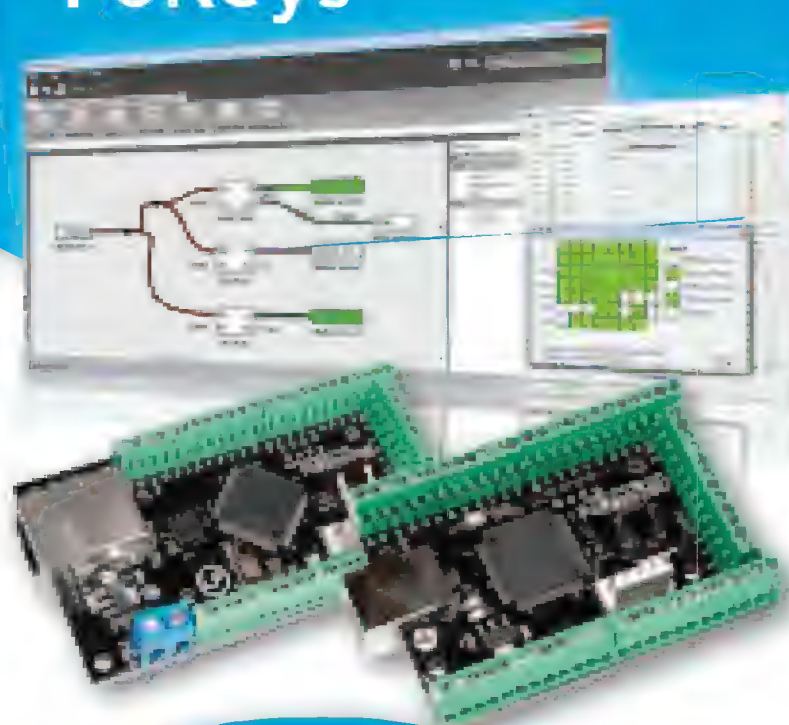


**INCREDIBLE LOW PRICES, FREE TECHNICAL SUPPORT
EXCELLENT CUSTOMER SERVICE WWW.SAELIG.COM**

Imagine this...

- a small and inexpensive
- USB or ethernet device
- with a rich set of features
- including PLC functionality and 3-axis stepper motor controller
- all accessible via free .NET, ActiveX or C++ library
- cross-platform
- configurable with free software

PoKeys



Or this...

- Smallest USB 2.0 portable 1MS/s oscilloscope
- Data acquisition of analog and digital signals
- Data recording
- Export to CSV, XLS, PDF and HTML
- Simple usage of advanced features
- Examples for C++, VB, Delphi and LabView
- Free software and updates

PoScope Mega1+



Polabs
Visit us at
www.poscope.com



Firgelli
www.firgelli.com

LINEAR SERVOS



L12-R Linear Servo

- Direct replacement for regular rotary servos
- Standard 3 wire connectors
- Compatible with most R/C receivers
- 1-2ms PWM control signal, 6v power
- 1", 2" and 4" strokes
- 3-10 lbs. force range
- 1/4" to 1" per second speed ranges
- Compatible with VEX



L16 Linear Actuators

New!

- 2", 4" and 6" strokes
- 10 - 40 lbs. force range
- 1/2" to 1" per second speed ranges
- Options include Limit Switches and Position Feedback

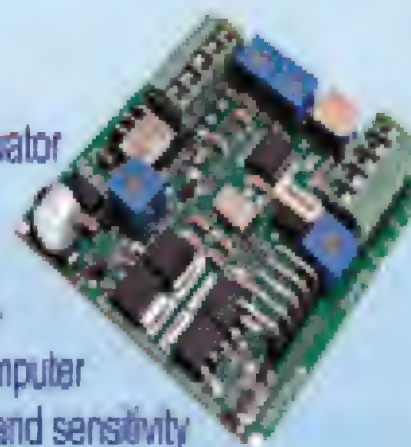
PQ12 Linear Actuator

- Miniature Linear Motion Devices
- 6 or 12 volts, 3/4" stroke
- Up to 5 lbs. force
- Integrated position feedback or limit switches at end of stroke
- External position control available



Linear Actuator Controller (LAC)

- Will drive any Linear Actuator with position feedback
- Up to 24v and 4 Amps
- USB connectivity to drive the actuator with your computer
- Adjustable speed, limits and sensitivity



L12-NXT Linear Servo

- Designed for LEGO Mindstorms NXT®
- Plugs directly into your NXT Brick
- NXT-G Block available for download
- Can be used with Technic and PF
- Max. speed: 1/2" per sec.
- Pushes up to 5 lbs.
- 2" and 4" strokes



Available Now @
www.firgelli.com

READER FEEDBACK

Continued from page 7

regulator as it did not require a heatsink. It puts out less heat and it's more efficient for a wide voltage input range. I ran the board for six months in varying temperature conditions (hot and cold) and not once did the component overheat, oscillate, or behave erratically. The switcher is also a drop-in equivalent to the LM78XX series in the TO220 case, so I did not have to change any components in my PCB program.

2. The two resistors in parallel were done for two reasons. The first is heat dissipation due to the voltage drop from 24 VDC to power the LEDs (2 VDC at 20 mA). This enables a complete parts list using only 1/4W resistors. Also note that the input voltage is 24 VDC – fairly common for industrial applications. Hobbyist applications typically run at 12 VDC. By removing one of the two resistors where they are paralleled and changing the output relay coil to a 12 VDC model, the board can now run on 12 VDC without any other modifications. So, the extra components add some design flexibility.

3. I have had a unit powered down for over three months and I just recently powered it back up. There was no loss in data and the unit picked right back up where it left off. This is a result of two things: the low current consumption of the time keeper chip and the storage capacity of the super capacitor. Based on some quick math, the data in the time keeper chip should be good for a year of storage. This is based on a RC time constant factoring in the chip voltage and current consumption combined with the 1.5 F capacitor.

Regards es 73 de,
Gerry Shand VE6BLI

Theremin Synthesizer Kit MkII

Create your own eerie science fiction sound effects by simply moving your hand near the antenna. Easy to set up and build. Complete kit contains PCB with overlay, pre-machined case and all specified components.

• PCB: 85 x 145mm
Cat. KC-5475

\$54.00*

POPULAR KIT



Test & Measure Kits

LED Battery Voltage Indicator Kit

This tiny circuit measures just 25mm x 25mm and will provide power indication and low voltage indication using a bi-color LED, and can be used in just about any piece of battery operated equipment. Current consumption is only 3mA at 6V and 8mA at 10V and the circuit is suitable for equipment powered from about 6-30VDC. With a simple circuit change, the bi-color LED will produce a red glow to indicate that the voltage has exceeded a preset value.

- PCB, bi-color LED and all specified electronic components supplied
- PCB: 25 x 25mm
Cat. KA-1778



\$7.25*

USB Power Monitor Kit

Plug this kit inline with a USB device to display the current that is drawn at any given time. Check the total power draw from an unpowered hub and its attached devices or what impact a USB device has on your laptop battery life. Displays current, voltage or power, is auto-ranging and will read as low as a few micro amps and up to over an amp. Kit supplied with double sided, solder masked and screen-printed PCB with SMD components pre soldered, LCD screen, and components.

- PCB: 65 x 36mm
Cat. KC-5516

Note: Laptop not included



\$43.25*

High Performance Timer Kit

This is a very sophisticated timer, which can be adapted for two different types of uses. The first is "one shot" operation, which can be used to keep electric windows active, or a thermo fan running for a period after ignition is switched off etc. The second is a pulse type operation, which can be used to squirt water spray for 1 second every 9 seconds. The uses are endless, and the time is adjustable from 0.1 seconds, to 16.5 minutes via easy to use (and accurate) digital switches. Kit supplied with PCB, and all electronic components.

- PCB: 105 x 60mm
Cat. KC-5379

\$31.00*

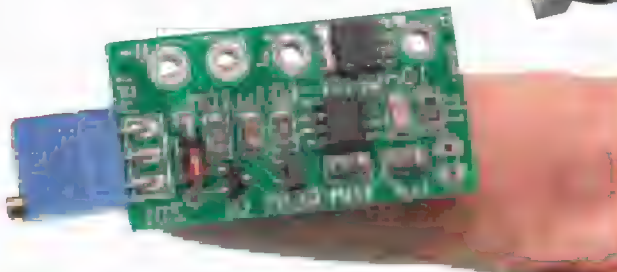


Battery Saver Kit

Cuts off the power between the battery and load when the battery becomes flat to prevent the battery over-discharging and becoming damaged. Suits SLA, Li-ion, Li-Po and LiFePO4 batteries between 6 to 24V. Uses very little power (<5uA) and handles 20A (30A peak). Supplied with double sided, solder masked and screen-printed PCB with SMDs pre-soldered (apart from voltage setting resistors) and components.

- PCB: 34 x 18.5mm
Cat. KC-5523

\$21.75*



NEW 2013

Universal Voltage Switch

A universal module suits a range of different applications. It will trip a relay when a preset voltage is reached. Can be configured to trip with a rising or falling voltage making it suitable for a wide variety of voltage outputting devices e.g. throttle position sensor, air flow sensor, EGO sensor. It also features adjustable hysteresis (the difference between trigger on/off voltage), making it extremely versatile. You could use it to trigger an extra fuel pump under high boost, anti-lag waste gate shutoff, and much more.

- Kit supplied with PCB, and electronic components.
- PCB size: 105 x 60mm
Cat. KC-5377

\$23.75*

POPULAR KIT



Jacob's Ladder High Voltage Display Kit MK2

With this kit and the purchase of a 12V ignition coil (available from auto stores and parts recyclers), create an awesome rising ladder of noisy sparks that emits the distinct smell of ozone. This improved circuit is suited to modern high power ignition coils and will deliver a spectacular visual display. Kit includes PCB, pre-cut wire/ladder and electronic components.

- 12V car battery, 7Ah SLA or > 5A DC power supply required
- PCB: 170 x 76mm
Cat. KC-5445

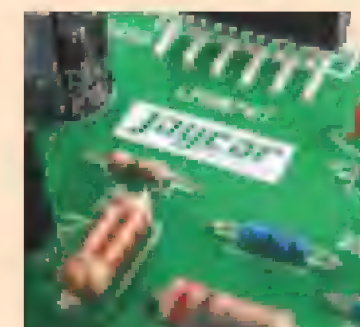
\$31.00*



ATTENTION KIT BUILDERS

Can't find the kit you are looking for? Try the Jaycar Kit Back Catalog

Our central warehouse keeps a quantity of older and slow-moving kits that can no longer be held in stores. A list of kits can be found on our website. Just search for "kit back catalog".



HOW TO ORDER

PHONE: 1800 784 0263*
FAX: +61 2 8832 3118*
EMAIL: techstore@jaycar.com
POST: P.O. Box 7172 Silverwater
DC NSW 1811 Australia

* Australian Eastern Standard Time
(Monday - Friday 6.30am - 5.30pm)

* US Eastern Standard Time
(Monday - Friday 4.30pm - 3.30am)

Prices valid until 30/11/2013

POST & PACKING CHARGES

Order value	Cost
\$25 - \$49.99	\$7.50
\$50 - \$99.99	\$20.00
\$100 - \$199.99	\$40.00
\$200 - \$499.99	\$60.00
\$500+	\$75.00

*ALL PRICES IN USD &
EXCLUDE POSTAGE & PACKING

Max weight: 12lb (5kg)
Heavier parcels: POA
Minimum order: \$25

Please note: Products are despatch from Australia,
so local customs duty & taxes may apply.

NOW shipping via



5-6 working days delivery

Don't Just Sit There...Build Something!

A New Airframe Design for Near Spacecraft: Part 1

One of the benefits of near space exploration is that it is a space science hobby on a shoestring budget. After recently having a capsule land in a tree and get damaged during its recovery process, I found I had the opportunity to replace the airframe with an even faster and easier design. Normally, I build airframes out of Styrofoam and surround them with an abrasion jacket and landing bag. That requires a lot of technical sewing which takes time. The new design does away with the vast majority of the sewing, and I believe results in a stronger airframe that I can more quickly complete.

But It's The Way I've Always Done It

My last near spaceflight tumbled pretty badly during its descent from near space. It was so bad, in fact, that it was unable to transmit a proper position report to the chase crew. As a result, we lost track of the near spacecraft and had to rely on a farmer to locate it. His extraction of the near spacecraft resulted in some damage to one of the airframes.

The damage was bad enough



This is an example of the near space airframes that I have been building since 1998. It's constructed from 3/4" thick Styrofoam and is wrapped in an abrasion jacket of blue ripstop nylon.

that I decided it was easier to replace the airframe rather than repair it. However, rather than make a new airframe the old way, I decided to design an airframe requiring less challenging techniques. That's what I will share this month.

A complete near spacecraft consists of a balloon, parachute, and a series of science and tracking modules. Many people construct the airframes of their modules from 3/4 inch thick Styrofoam. This Styrofoam is the blue or pink sheets used to insulate the walls of homes and not the white beady stuff.

The Styrofoam is lightweight and strong enough to contain the avionics once the Styrofoam sheets are hot

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?/magazine/article/november2013_NearSpace.

glued together. However, the Styrofoam airframe is not strong enough to attach directly to the parachute. That's why I make an abrasion jacket from ripstop nylon to place the Styrofoam box inside. The parachute shroud lines are attached to four straps sewn to the top of the abrasion jacket.

The abrasion jacket has two other important functions. First, it protects the Styrofoam body of the airframe from bumps and scrapes during landing. Second, it adds color to the airframe which helps it stand out on the ground. The color — if it's dark — has a second positive effect by absorbing solar radiation. It gets cold in near space; down to -60 degrees Fahrenheit on a good day. Any solar absorption by the abrasion jacket helps keep the interior of the airframe a bit warmer during near space missions.

Now, the negative side of the abrasion jacket: they're difficult to sew. That's because there are a lot of corners in the jacket that have to get sewn into. It's much easier to sew a parachute even though it's much larger because it's round and doesn't contain the corners found in an abrasion jacket.

So, if I was going to design an easier to construct airframe, I would have to replace the abrasion jacket and find a new way to attach the parachute and other modules. The result I came up with is a design that I believe is stronger because of the way the airframe's straps are attached.

The Newer Design

I began by cutting out six pieces of Styrofoam that when assembled, would be a cube 10 inches on a side. The Styrofoam sheets forming the top and bottom faces of the cube measured 10 inches square, while the sides measured 8.50 inches tall and 9.25 inches wide. I then glued the four sides and bottom piece together with hot glue.

In a departure from my traditional design, I strengthened the inside corners of the box with triangular strips of Styrofoam. The strips extend from the bottom of the airframe and reach to within 3/4 inches of the top. The 3/4 inch gap at the top of the strips left space for the hatch that closes the top of the airframe. At this point, I had a 10 inch cube of Styrofoam missing its lid.

After the glue cooled, I shaved a one inch wide strip off the corners of the airframe. This did not weaken the airframe because of the extra Styrofoam glued to the inside corners of it. I then cut a square opening in each side of the airframe. Three of the openings are for the experiments and antenna booms that I'll bolt to the airframe, and the fourth is for the flight computer's control panel.

After cleaning up the loose



This is what the airframe looked like after gluing five of the sides together. Note that I beefed up the interior corners of the airframe with long wedges of Styrofoam.

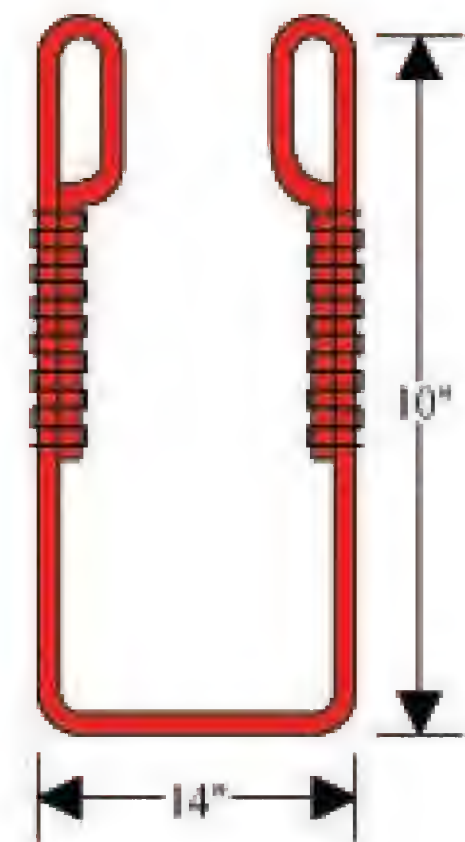
Styrofoam dust, I wrapped the outside of the airframe in colored Mylar tape; black, in this case. The tape contains an acrylic adhesive and is used by mail order companies to seal and color-code packages. It's also the same tape used to give Styrofoam gliders color and protection from bumps. The tape is available from Uline as model number S-700. I chose black because the color increases the amount of solar radiation collected by the airframe during its near space mission.

Now, it was time to assemble the harness for the airframe which replaced the abrasion bag. The harness was to be a pair of thick nylon straps surrounding the Styrofoam airframe. They support the weight of the airframe and are where I would connect the parachute to the airframe. The harness would also permit the recovery parachute and other modules to connect to the airframe without their applied forces cracking the Styrofoam walls.

I cut two strips of one inch wide nylon ribbon 32 inches long. Before any sewing, I took a hot soldering iron and melted the cut edges of the nylon ribbon. Melting nylon like this prevents the nylon fibers in the strap from unraveling. I don't recommend using your best soldering iron for this, however. It's also a good idea to perform this operation in a location with good ventilation. After melting the ends of the straps, I doubled over four inches of strap on both ends and sewed them together to create a 1/2 inch open loop at their tops.

I next sewed the two straps together at their centers to form the cross-shaped harness. I planned to set the airframe over the center of the harness and then pull the arms of the

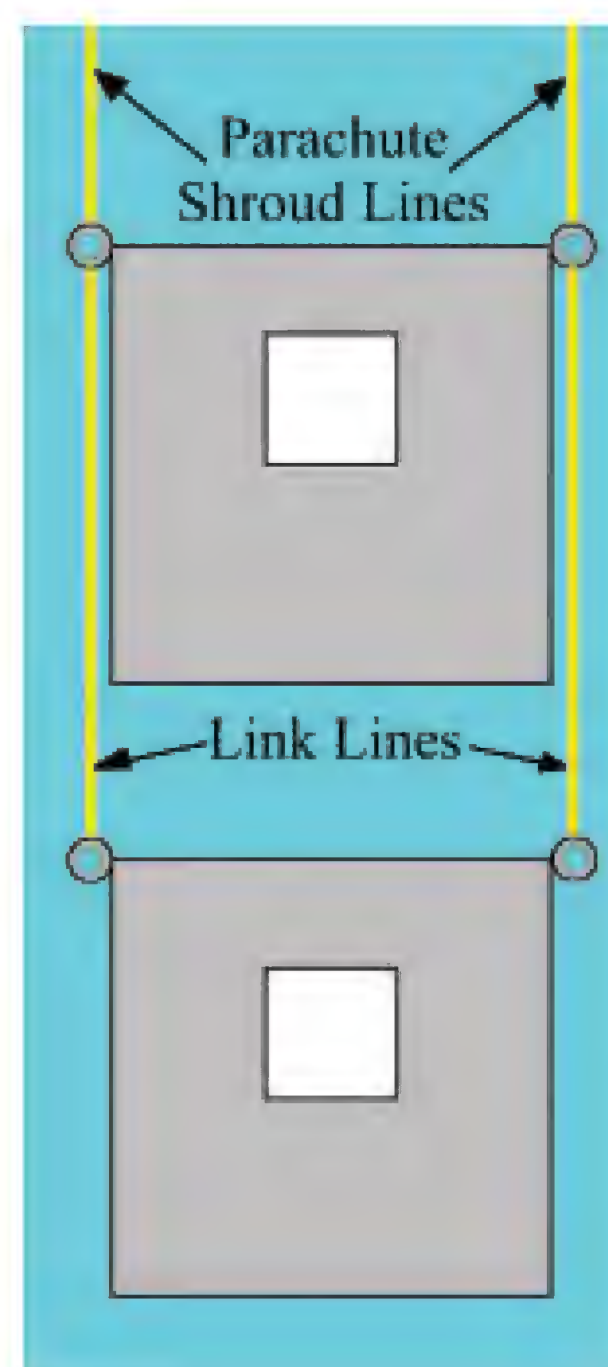
The airframe uses a harness of two nylon straps. The tops of the harness are folded over and sewn together to create a half inch loop.



harness up the corners of the airframe. The loops sewn to the top of the harness would be at the top of the airframe; this is where I insert split rings (the type used in key rings). The parachute's shroud lines would then connect to the airframe at these split rings.

In my old capsule design, I sewed loops and split rings to the top and bottom of the abrasion jacket. The top split ring supported the force of the parachute while the bottom split rings supported the weight of the modules below. This created tension on the abrasion jacket from the parachute pulling up and the modules pulling down.

That tension was strong enough



Because of weight limits imposed by FAR 101, amateur near spacecraft typically consist of two or more modules tethered together. In NearSys missions, link lines tether the modules together. In order to reduce the stress acting on the abrasion jacket, the parachute's shroud lines and the link lines attach to the same split rings at the top of the airframe. It's easier to replace a warped split ring than to re-sew a torn abrasion jacket, should the stress from the parachute pulling up and the weight of the lower module pulling down gets a little too strong.

to tear a couple of Dacron straps off the jacket. After the second loop was torn, I stopped attaching the modules to the bottom split rings and began attaching both the parachute and lower modules to the split rings at the top of the airframe. Four lift lines — or short lengths of woven Dacron kite line — span the distance between the lower module and the split rings.

After switching to using only the top split ring to tether parachutes and lower modules, I didn't do away with the bottom split rings. I repurposed them. I run the link lines from the lower module through the bottom split rings without actually attaching the link lines to the bottom split ring.

The bottom ring's small interior diameter seems to prevent the airframes from swinging wildly like a pendulum. I like this so much that in the new design, I created restraining loops at the bottom of the harness. In this case, however, I added four loops of Dacron kite line to the harness. On the harness at a point one inch above the bottom of the airframe, I melted a hole through all four arms of the harness. To strengthen the hole for the Dacron loop, I inserted an opened rivet

The largest and oldest conference for amateur near space explorers is the Great Plains Super Launch (GPSL). It takes place every summer in a location in the Midwest. The 2014 Great Plains Super Launch will take place in Hutchinson, KS — the home of the second largest air and space museum in the United States: the Cosmosphere. The dates for the conference are June 12 and 13, with the 12th set aside for presentations and the 13th reserved for near space launches. If you would like to begin your own program or investigate how and what near space explorers do, then plan to attend the next GPSL. While you're there, be sure to look for me. By the way, I want to give Zack a big hand for hosting GPSL again.

through the hole. Then, I cut a short length of woven Dacron kite line and melted the cut ends with a lighter to prevent the kite line from unraveling. I tied a loop with the kite line by folding it in half and tying a simple overhand knot in the end.

Next, I pulled the folded end of the Dacron kite line loop through the center of the rivet using a pair of tweezers. The knot at the other end of the loop is so large that it can't be pulled through the rivet. The loop of Dacron left sticking out of the rivet replaces the split ring I used to restrain the swinging of modules.

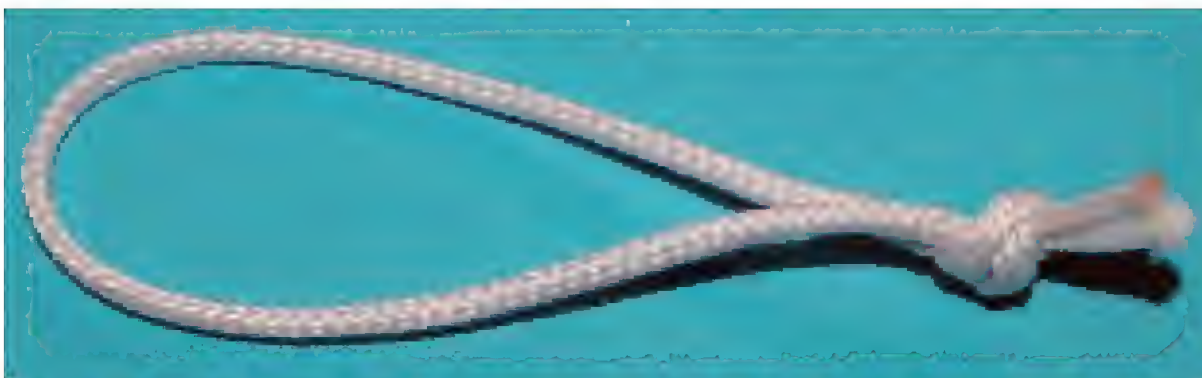
Before attaching the harness to the airframe, I added a bumper to the bottom of the airframe. My older modules had a landing bag at the bottom of the airframe. The landing bag protected it in case it landed on a hard surface like a road. The landing bag consisted of a one inch thick sheet of foam rubber placed between the Styrofoam bottom of the airframe and the inside of the

bottom of the abrasion jacket.

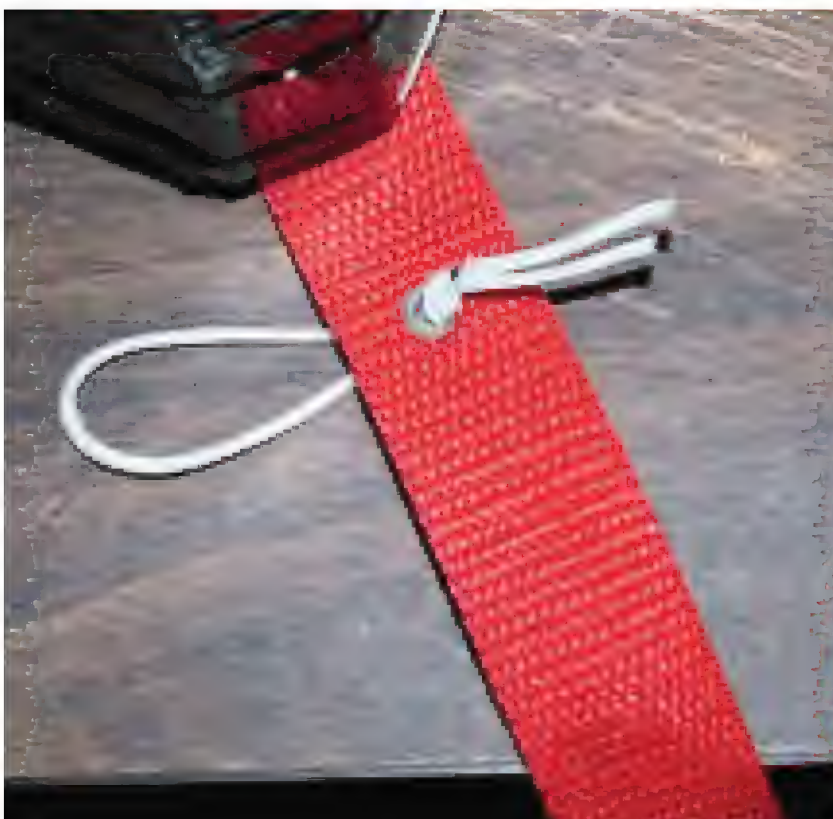
I took a hint from hard hats when I replaced the landing bag in the new design. Hard hats protect heads two different ways. First, the elastic band inside of them spreads out an impact's length of time, thereby reducing the force of the peak impact. That's how the landing bag functioned in my previous airframe design.

Second, hard hats have a hard shell that spreads out the area of the force of an impact, thereby reducing the peak pressure of the impact. I chose the second method to protect the airframe in the new design.

I spread out the area of the landing impact by adding a lightweight plastic plate to the bottom of the airframe. The plastic was a 1/4 inch thick sheet of Coroplast, or corrugated polypropylene plastic. This impact bumper attaches to the bottom of the airframe with bolts that also attach the harness to the bottom of the airframe.



A six inch loop of Dacron was folded in half and tied at the end with a simple overhand knot. Note that the cut ends of the kite line are melted to prevent them from unraveling over time.



The knot in the Dacron loop will remain inside the rivet after the harness arms are attached to the sides of the airframe.



In this bottom view of the airframe, you can see the impact bumper attaches to the bottom of the airframe using the same bolts that attaches the four arms of the harness to the bottom of the airframe.

Wow! It's been 10 years since I started writing for *Nuts & Volts*. This will be the 64th article I have written about near space exploration. I've focused primarily on the science technology aspects of this hobby. One result was that it gave me the idea and confidence to base my PhD dissertation around BalloonSats and its effects on student interest in science. Thanks *Nuts & Volts*.

I temporarily taped the impact bumper and harness to the bottom of the airframe and then drilled four holes through the straps, plate, and airframe. I then used four #6-32 bolts, fender washers, and nylocks to attach the harness and impact bumper to the bottom of the airframe.

Now, it was time to attach the arms of the harness to the airframe. So, I drilled two holes through each harness strap and corner. Before inserting bolts through the holes in the straps, I ran a hot soldering iron through the holes to melt their edges. Sixteen #6-32 bolts, 16 fender washers, and eight nylocks bolt the sides of the airframe to the harness.

The fender washers are important because they increase the surface area under compression by

the nylocks. Spreading out the area under pressure lessens the likelihood that the Styrofoam will crack from the compression.

That completed the construction of the airframe. The next step was to modify the avionics pallet, battery box, and hatch closure. I'll write about those modifications next time.

The new airframe is a bit heavier than my older design. However, I feel this new design is stronger and addresses weakness issues in the older design. The old design was good and rarely experienced a failure. I'm hoping to reduce those rare failures to the point where I will never see them again.

When the chaos of burst gets really rough, the near spacecraft will continue to transmit position reports

I used two bolts to attach each arm of the harness to the airframe. The fender washers are important because without them the compression from the nylocks will crush the Styrofoam of the airframe.



that chase crews can use.

Onwards and Upwards,
Your near space guide **NV**

Be a part of the Institute of Navigation's Annual Autonomous Snowplow Competition January 23 - 26, 2014 Saint Paul, Minnesota

A competition devoted to the art of navigation

Be a spectator, a sponsor or a competitor!

To learn more visit: www.autosnowplow.com or www.ion.org



Sponsored by



Follow Us On



@plowbot



Autonomous Snowplow Competition



Auto Snowplow

ELECTRONET

HOBBYENGINEERING

Kits, Parts and Supplies
www.HobbyEngineering.com

GET THE NUTS&VOLTS DISCOUNT!
 Mention or enter coupon code **NVRMZ12**
 and receive 10% off your order!

ramsey
www.ramseykits.com
 AM/FM Broadcasters • Hobby Kits
 Learning Kits Test Equipment
 ...AND LOTS OF NEAT STUFF!

USB Add USB to your next project--
 It's easier than you might think!

DLP Design
 USB-FIFO • USB-UART • USB/Microcontroller Boards
 RFID Readers • Design/Manufacturing Services Available
Absolutely NO driver software development required!
www.dlpdesign.com

Pololu
 Robotics & Electronics
WWW.POLOLU.COM

superbrightleds.com
 COMPONENT LEDs • LED BULBS • LED ACCENT LIGHTS

THE ORIGINAL SINCE 1994
PCB-POOL
 Beta LAYOUT
 • Low Cost PCB prototypes
 • Free laser SMT stencil
 with all Proto orders
WWW.PCB-POOL.COM

Aztec
 MCU Prototyping
 PICAXE and Arduino design and development tools
www.aztecmcu.com

Serial LCD & LED
www.melabs.com

FREE PCB Design Software

Get Started Now
 at FabStream.com

FabStream
 A Comprehensive Technology Provider

www.canakit.com

- Over 200 Kits & Modules to choose from!
- Easy Worldwide Online Ordering
- Fast Delivery & Same Day Shipping

FM Transmitters • Timers • Audio Amplifiers • Motor Controllers

Karlsson Robotics

For Electronic Components, SMT Adapters,
 Modules, Custom Design, Assembly and Prototyping.

www.kr4.us

For the ElectroNet
 online, go to
www.nutsvolts.com
 click **Electro-Net**

ALL ELECTRONICS CORPORATION
 Electronic Parts
 and Supplies.
www.allelectronics.com
 Free 96 page catalog 1-800-826-5432

CORIDIUM
 ARM controllers running
 floating point **BASIC** from \$10.00
www.coridium.us

INVEST in your BOT!

HITEC

12115 Paine Street • Poway, CA 92064 • 858-748-6948 • www.hitecrd.com

MODERN DEVICE

JeeLabs - wireless + Arduino
 on a budget

JeeNode SMD \$20

8888 thr-hole Arduino kit	\$13
LCD117 Serial LCD board	\$10
Heartbeat Sensor	\$18
Wind Sensor	\$17
Fluxamsynth Synth Shield	\$35

use code 74V for a 10% discount*
moderndevice.com *30 days

2-For-1 Holiday Offer

Give a one year (12 issues)
subscription to a friend and
receive a **FREE** one year
subscription for another
friend, or yourself!



Not only will they get a full year of Nuts & Volts delivered right to their door, they'll also get free access to our digital (web) edition, PLUS our mobile (iOS, Android, Kindle Fire) edition.

Subscribe online at:
www.nutsvolts.com

or call:

877-525-2539 (toll free)

818-487-4545 (outside US)

Be sure to use promotion code
Y3WXMS when ordering

U.S. Price:
1 Yr - \$26.95

The Nuts & Volts **WEBSTORE**

GREAT FOR DIYers!

Programming Arduino Next Steps: Going Further with Sketches by Simon Monk

NEW!

Arduino guru Simon Monk reveals advanced programming techniques for Arduino! *Programming Arduino Next Steps: Going Further with Sketches* is the must-have follow-up to Monk's bestseller, *Programming Arduino: Getting Started with Sketches*. Aimed at experienced programmers and hobbyists who have mastered the basics, this book takes you "under the hood" of the Arduino, revealing professional-level programming secrets.

\$20.00

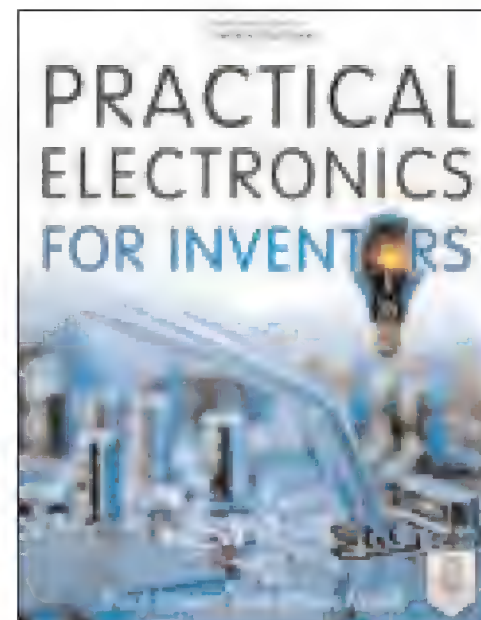


Practical Electronics for Inventors, Third Edition by Paul Scherz, Simon Monk

The electronics know-how you need to become a successful inventor.

Spark your creativity and gain the electronics skills required to transform your innovative ideas into functioning gadgets. This hands-on updated guide outlines electrical principles and provides thorough easy-to-follow instructions, schematics, and illustrations.

\$39.95

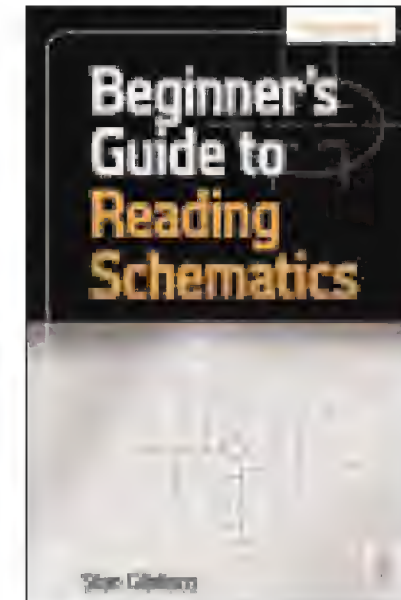


Beginner's Guide to Reading Schematics, 3E by Stan Gibilisco

NEW!

Navigate the roadmaps of simple electronic circuits and complex systems with help from an experienced engineer. With all-new art and demo circuits you can build, this hands-on, illustrated guide explains how to understand and create high-precision electronics diagrams. Find out how to identify parts and connections, decipher element ratings, and apply diagram-based information in your own projects.

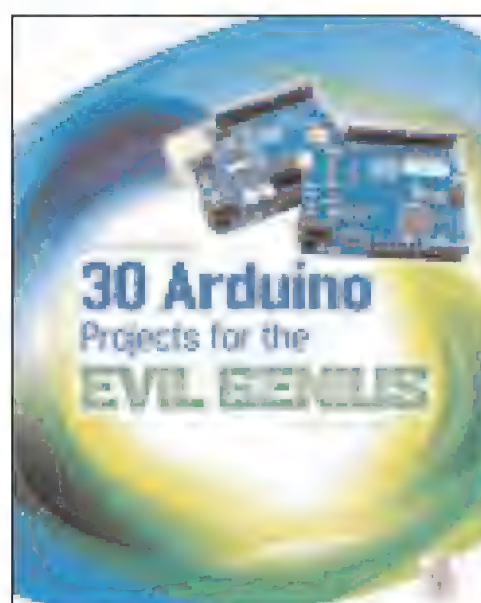
\$25.00



30 Arduino Projects for the Evil Genius: Second Edition by Simon Monk

Fully updated throughout, this do-it-yourself guide shows you how to program and build fascinating projects with the Arduino Uno and Leonardo boards, and the Arduino 1.0 development environment. *30 Arduino Projects for the Evil Genius, Second Edition*, gets you started right away with the simplified C programming you need to know, and demonstrates how to take advantage of the latest Arduino capabilities.

\$25.00



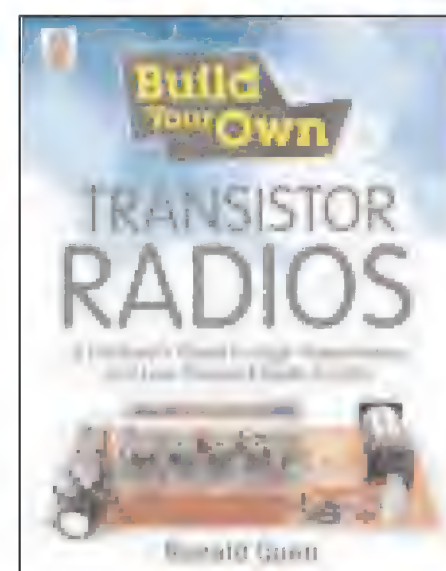
Build Your Own Transistor Radios by Ronald Quan

A Hobbyist's Guide to High Performance and Low-Powered Radio Circuits

Create sophisticated transistor radios that are inexpensive yet highly efficient. Inside this book, it offers complete projects with detailed schematics and insights on how the radios were designed. Learn how to choose components, construct the different types of radios, and troubleshoot your work.

***Paperback, 496 pages**

\$49.95



Raspberry Pi Projects for the Evil Genius by Donald Norris

This wickedly inventive guide shows you how to create all kinds of entertaining and practical projects with the Raspberry Pi operating system and programming environment. Each fun, inexpensive Evil Genius project includes a detailed list of materials, sources for parts, schematics, and lots of clear, well-illustrated instructions for easy assembly. The larger workbook-style layout makes following the step-by-step instructions a breeze.

\$25.00



How to Diagnose and Fix Everything Electronic by Michael Jay Geier

Master the Art of Electronics Repair

In this hands-on guide, a lifelong electronics repair guru shares his tested techniques and invaluable insights. *How to Diagnose and Fix Everything Electronic* shows you how to repair and extend the life of all kinds of solid-state devices, from modern digital gadgetry to cherished analog products of yesteryear.

\$24.95



Programming PICs in Basic by Chuck Hellebuyck

If you wanted to learn how to program microcontrollers, then you've found the right book! Microchip PIC microcontrollers are being designed into electronics throughout the world and none is more popular than the eight-pin version. Now the home hobbyist can create projects with these little microcontrollers using a low cost development tool called the CHIPAXE system and the Basic software language. Chuck Hellebuyck introduces how to use this development setup to build useful projects with an eight-pin PIC12F683 microcontroller.



\$14.95

Master and Command C for PIC MCUs by Fred Eady

Master and Command C for PIC MCU, Volume 1 aims to help readers get the most out of the Custom Computer Services C compiler for PIC microcontrollers.

The author describes some basic compiler operations that will help programmers particularly those new to the craft create solid code that lends itself to easy debugging and testing. As Eady notes in his preface, a single built-in CCS compiler call (output_bit) can serve as a basic aid to let programmers know about the "health" of their PIC code.

\$14.95



Order online @ www.store.nutsvolts.com
Or CALL 1-800-783-4624 today!

BOOK & KIT COMBOS

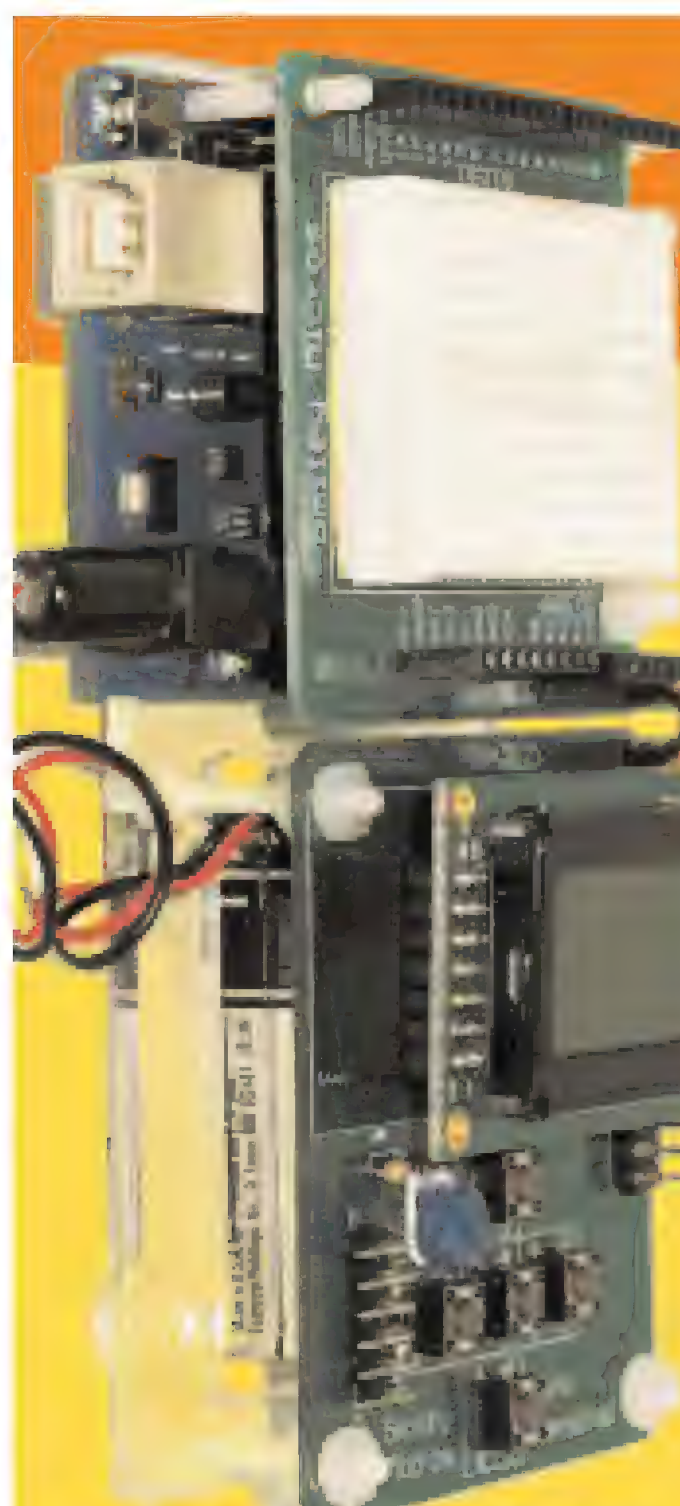
16-Bit Complete Combo



Beginner's Guide Complete Combo!



Arduino Handheld Prototyper System



Arduino Proto Shield
\$16.99

I2C Mini Terminal
\$29.99

Handheld Base
\$13.99

All three*: \$59.99

*Arduino and battery not included

From Smiley's Workshop

CD-ROM SPECIAL

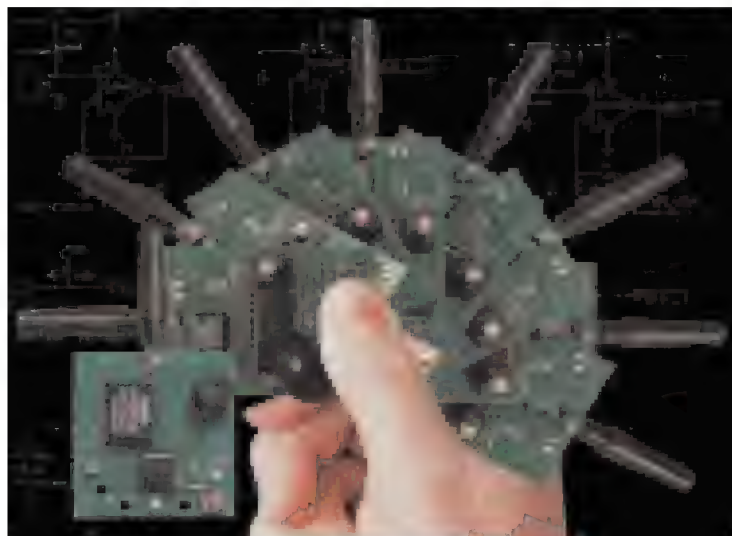
Nuts & Volts
9 CD-ROMs
& Hat Special!
That's 108 issues.
Complete with supporting
code and media files.



POCKET REF
The Nuts & Volts
Pocket Ref
All the info you need at your fingertips!
This great little book is a concise all-purpose reference featuring hundreds of tables, maps, formulas, constants & conversions. AND it still fits in your shirt pocket!
Only \$12.95
Visit <http://store.nutsvolts.com> or call (800) 783-4624

PROJECTS

Super Detector Circuit Set



Pick a circuit!
 With one PCB you have the option of detecting wirelessly:
 temperature, vibration, light, sound, motion, normally open switch, normally closed switch, any varying resistor input, voltage input, mA input, and tilt, just to name a few.

Subscriber's Price \$32.95
Non-Subscriber's Price \$35.95

3D LED Cube Kit



This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes. Colors available: Green, Red, Yellow & Blue

Subscriber's Price \$57.95
Non-Subscriber's Price \$59.95

Radiation Monitor Alarm Kit



This is an inexpensive surface-mount project that is good for beginners to start with. This kit has its own printed circuit board (PCB) which makes mounting the components easy. Plus, it comes in a pocket size shielded aluminum case measuring 3.5" in length and 1" in diameter. The on-off switch is a pushbutton on the bottom.

Subscriber's Price \$32.45
Non-Subscriber's Price \$33.95

Geiger Counter Kit

As seen in the March 2013 issue.

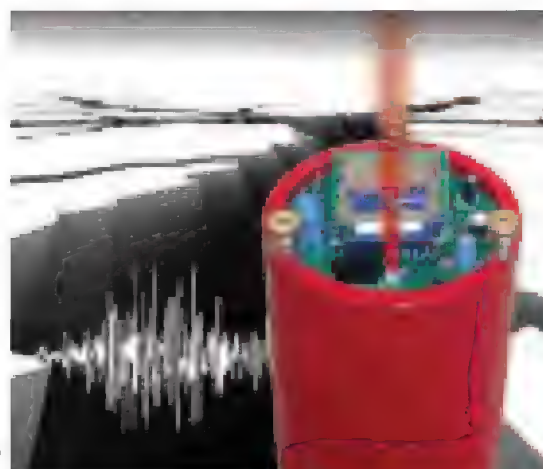


This kit is a great project for high school and university students. The unit detects and displays levels of radiation, and can detect and display dosage levels as low as one micro-roentgen/hr. The LND712 tube in our kit is capable of measuring alpha, beta, and gamma particles.

Partial kits also available.
Subscriber's Price \$145.95
Non-Subscriber's Price \$148.95

Seismograph Kit

As seen in the May 2012 issue.



Now you can record your own shaking, rattling, and rolling.

The Poor Man's Seismograph is a great project /device to record any movement in an area where you normally shouldn't have any. The kit includes everything needed to build the seismograph. All you need is your PC, SD card, and to download the free software to view the seismic event graph.

Subscriber's Price \$79.95
Non-Subscriber's Price \$84.95

Magic Box Kit

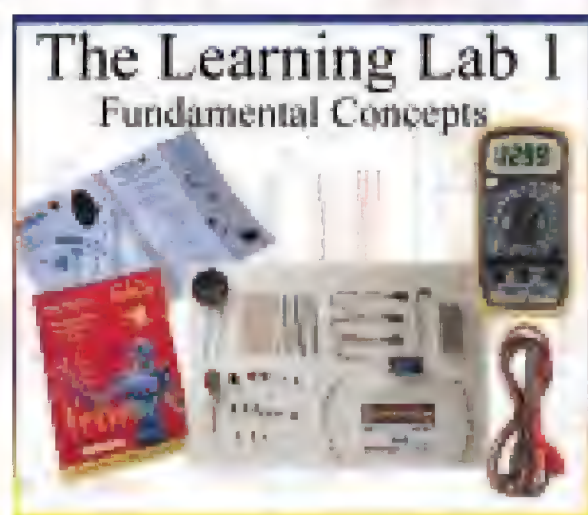
As seen in the November 2011 issue.



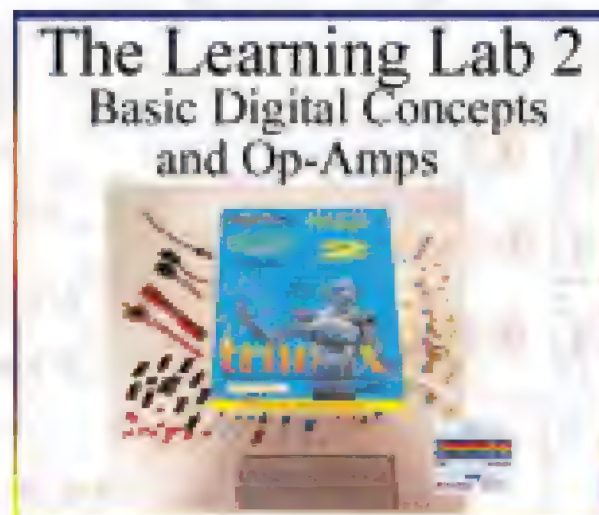
Battery Marvel helps protect cars, trucks, motorcycles, boats, and any other 12V vehicles from sudden battery failure. This easy-to-build kit features a single LED that glows green, yellow, or red, indicating battery health at a glance. An extra-loud piezo driver alerts you to any problems.

Details, please visit our website.
Subscriber's Price \$18.95
Non-Subscriber's Price \$19.95

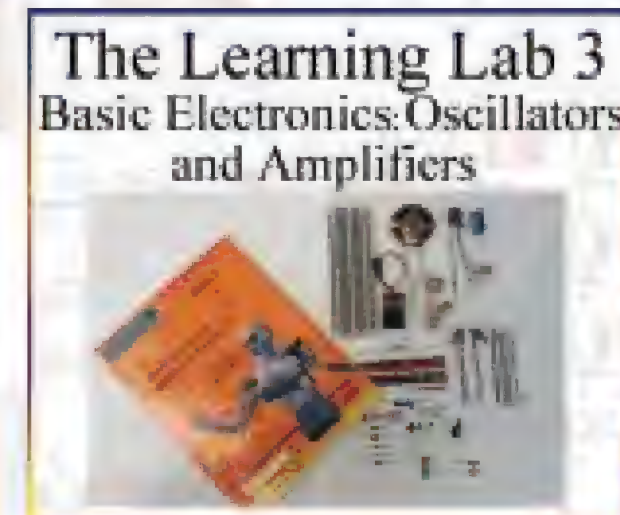
FOR BEGINNER GEEKS!



\$59.95



\$49.95



\$39.95

The labs in this series — from GSS Tech Ed — show simple and interesting experiments and lessons, all done on a solderless circuit board.

As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info and a promotional video, please visit our webstore.

Discovering a Better and More Powerful Engineering Workstation

Post comments on this article and find any associated files and/or downloads at www.nutsvolts.com/index.php?/magazine/article/november2013_DesignCycle.

My very first piece of Heathkit test equipment was an oscilloscope. It was quickly followed by my second Heathkit, which was an audio waveform generator. EICO was also selling hobby and test equipment kits in those days. My first meter — as we called them then — was realized through an EICO 232-K VTVM (Vacuum Tube Volt Meter) kit. I didn't stop with test equipment. As a teenager with very limited funds, I talked my dad into a Heathkit bass guitar amplifier kit, which we later upholstered with dad's favorite orange furniture covering. With two 12 inch speakers, my Heathkit T-38 bass guitar used to regularly rock the neighborhood.

I'm sure that many of you have built a Heathkit in some form or fashion. I recall Heathkit being big into color television kits. I wasn't brave enough (or rich enough) to fly at that level. If you were into electronic repair, you probably owned or knew someone that owned an EICO tube tester. Unfortunately, all of those must-have pieces of test equipment have gone the way of the dinosaurs.

No Rocking Chair for Analog Just Yet

Note that every piece of Heathkit and EICO test gear I mentioned operated in the analog domain. Although vacuum tubes were among the very first components found in "digital" devices of the day, there's really nothing inherently digital about a vacuum tube. Unless you were a ham operator, your interest in tubes was audio or television based.

The use of vacuum tubes has regained its acceptance in the world of guitar amplification. However, you won't find any tubes in today's computing devices. Digital has taken front row status in computing, portable audio devices, and video. However, you as a human can't hear and understand digital. I remember sitting in a local RadioShack and watching guys come in and load their programs into the demo TRS-80s via a cassette. They would ask for a cheap transistor radio they would use to "listen" to the electrical noise generated by the TRS's analog-to-digital converter (ADC) circuitry.

When they put the transistor radio in close proximity of the TRS-80 and heard this distinct electrical noise, they knew that audio tones on the cassette were most likely being converted to digital signals the TRS-80 could use.

To "hear" and "see" digital signals one may think we would need equipment similar to the Heathkits and EICO kits I mentioned earlier. In reality, we do. However, bring a few thousand dollars with you because today's test equipment is expensive.

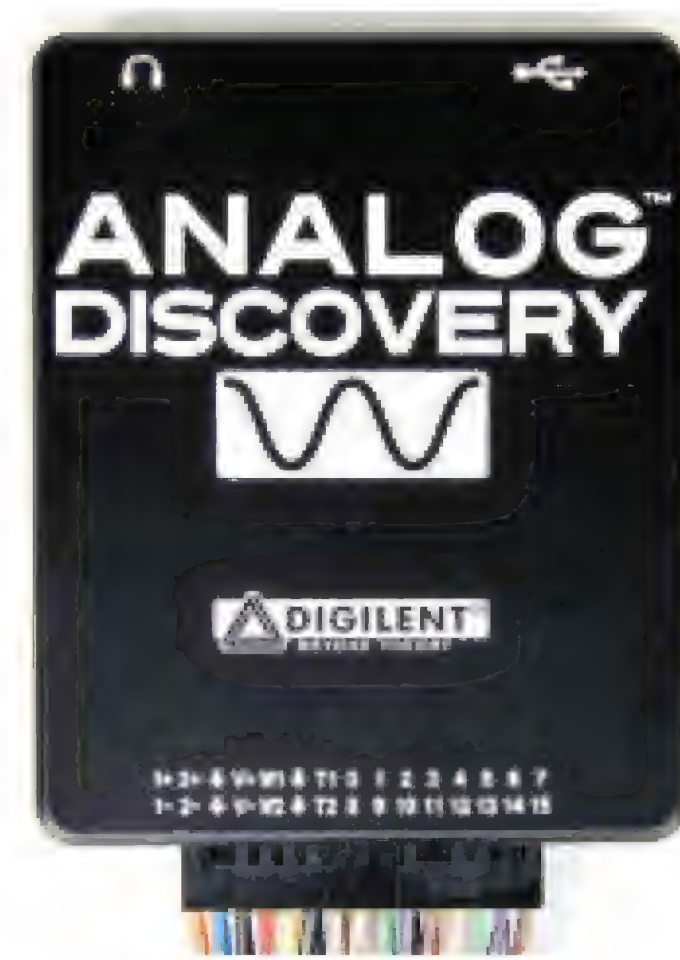
The good news is that with

today's digital technology one can pack an oscilloscope, signal generator, and voltmeter in the space of a pack of cigarettes. Plus, you'll only spend a couple of hundred dollars instead of a couple of thousand dollars.

The Analog Discovery

The little black box you see in **Photo 1** contains a two-channel oscilloscope, a two-channel arbitrary waveform generator (AWG), a 16-bit digital logic analyzer, and a 16-bit digital pattern generator. The Analog Discovery from Digilent is powered from a host USB portal and internally generates ± 5 VDC power rails, which are available to the user. Up to 16 bits of GPIO are available in addition to a voltmeter, a spectrum analyzer, and a network analyzer.

If you are reading this, you most likely own a PC. All you need to use the Analog Discovery to its full



■ **Photo 1.** The Analog Discovery is a mixture of Xilinx FPGA technology, an Analog Devices dual-channel analog-to-digital converter, and an Analog Devices dual-channel digital-to-analog converter.



■ Photo 2. The Analog Discovery is easily connected to the target circuitry via the 30-pin fly-wire interface. All of the physical features of the Analog Discovery are pinned out to the fly-wire.



■ Photo 3. This seemingly simple piece of optional Analog Discovery equipment is a force multiplier. The Discovery BNC allows the use of standard oscilloscope probes and provides the option to enable a standard 50Ω test lead output impedance.

■ Photo 4. The Analog Discovery fully decked out and ready for action.



potential is a PC and a solderless breadboard. You already know what the breadboard is for. The PC's job is to power the Analog Discovery, and communicate with it using the Waveforms application.

The Analog Discovery is really a teaching tool disguised as a bench instrument. The idea behind the Discovery is to allow the user to doodle with analog and digital circuitry anywhere at any time, and document the results. Many teaching professionals are utilizing the Discovery in the classroom. The students benefit by being able to take the “lab” home or back to their dormitory room. All of the Analog Discovery's graph and display data can be captured and exported for further analysis.

The Discovery interfaces to the outside world through a 2x15 fly-wire. The fly-wire and pin assignments are under the camera lens in **Photo 2**. The differential oscilloscope channels are marked as ± 1 and ± 2 . A pair of ground pins separates the scope inputs from the ± 5 VDC power pins and arbitrary waveform generator fly-wires. A second pair of ground pins delineates the triggers and 16 bits of GPIO. A very nice color graphic detailing the fly-wire layout can be had from the Digilent website. There may be instances in which the fly-wire arrangement just doesn't cut it for one reason or another. Not to worry. Just pull out that set of oscilloscope probes and put the device shining in **Photo 3** to work. The Discovery BNC breaks out the oscilloscope and arbitrary waveform generator channels to BNC connectors. All of the remaining signals are passed through pin for pin in fly-wire order.

As you can see in **Photo 3**, the oscilloscope BNC terminations can be DC or AC coupled, depending on the placement of a couple of jumpers. The arbitrary waveform generator outputs can be configured with jumpers to present a 0Ω or 50Ω output impedance. The Discovery BNC attaches directly to the Analog Discovery as shown in **Photo 4**. The hardware architecture of the Analog Discovery allows each instrument to stand independently or share resources with other instruments. I took a peek under the covers of my Analog Discovery. It is primarily made up of a Xilinx XC6SLX16 Spartan 6 FPGA, a dual-channel AD9717 Analog Devices digital-to-analog converter (DAC), and a dual-channel Analog Devices AD9648 ADC. An FTDI FT232 USB-to-serial IC handles the Analog Discovery-to-PC interconnect.

Naturally, the Analog Discovery circuitry also includes various supporting op-amps, clocks, and amplifiers. To my knowledge, there is no publically available schematic. That's okay because even the most advanced hobbyist would have a hard time scratch building or even repairing certain areas of the Analog Discovery printed circuit board (PCB).

Scoping a Big Muff Pi

If you play electric guitar, you know all about the Big Muff Pi sitting in **Photo 5**. If you are into bands like Pink Floyd and Santana, you've heard David Gilmour and Carlos Santana play through this device. The Big Muff Pi is a fuzzbox. A fuzzbox introduces extreme distortion to the original input signal.

The best way to describe a fuzz tone is to see one. I attached the Discovery's first oscilloscope input to the input jack of the Big Muff Pi. The second oscilloscope input performed guard duty at the Big Muff Pi's output jack. The input signal is a 500 Hz sine wave courtesy of the Analog Discovery's first arbitrary waveform generator. As you can see in **Screenshot 1**, the input sine wave amplitude is 50 mV. Note that the original sine wave is presented at the output of the Big Muff Pi slightly distorted and out of phase. The Big Muff Pi controls at this point are at their minimum positions.

I chose to add some measurements in the upper right of the oscilloscope view you see in **Screenshot 1**. You can also see the oscilloscope's time base, channel 1, and channel 2 settings in this area of **Screenshot 1**.

Let's turn the Big Muff Pi's control potentiometers to their maximum positions. What you see in **Screenshot 2** is fuzz. Not only did the Big Muff Pi distort the input sine wave, it also amplified it. I activated the Discovery's oscilloscope persistence feature to enhance the trace view.

Virtual Switches and Such

Everyone has that one thing they really just don't enjoy doing unless it is absolutely necessary. For me, it's mounting switches and LEDs on solderless breadboards. It seems that the switch terminations are always just a wee bit too short or too large for the breadboard holes. That's why one of my favorite features of the Discovery is the virtual pushbutton/switch/LED options that can be applied to its GPIO.

In **Screenshot 3**, I've electrically connected GPIO 0 to GPIO 8, and GPIO 1 to GPIO 9 using the fly-wires and a male header. Obviously, the GPIO 0 position is configured as a pushbutton. GPIO is a simple on/off slide switch. GPIO 2 is configured as a logic source that includes a logical high, logical low, and high impedance state. The remaining GPIOs are breakouts of the GPIO 2 logic switch. I left GPIOs 8 through 15 in their default LED configurations. We will use LEDs 8 through 15 as indicators.

The logic levels of all the switches are displayed on their personal LED indicators. This is done to eliminate the need of wasting a GPIO just to indicate a logic level. To demonstrate the logic level output of the switches at GPIO positions 0 and 1, I tied their outputs to the LEDs at GPIO positions 8 and 9, respectively. As you can see, I've clicked on the pushbutton and set the output at GPIO 1 to logically high. The GPIO 8 and GPIO 9 LEDs and the switch 0 and switch 1 personal LEDs reflect those actions.

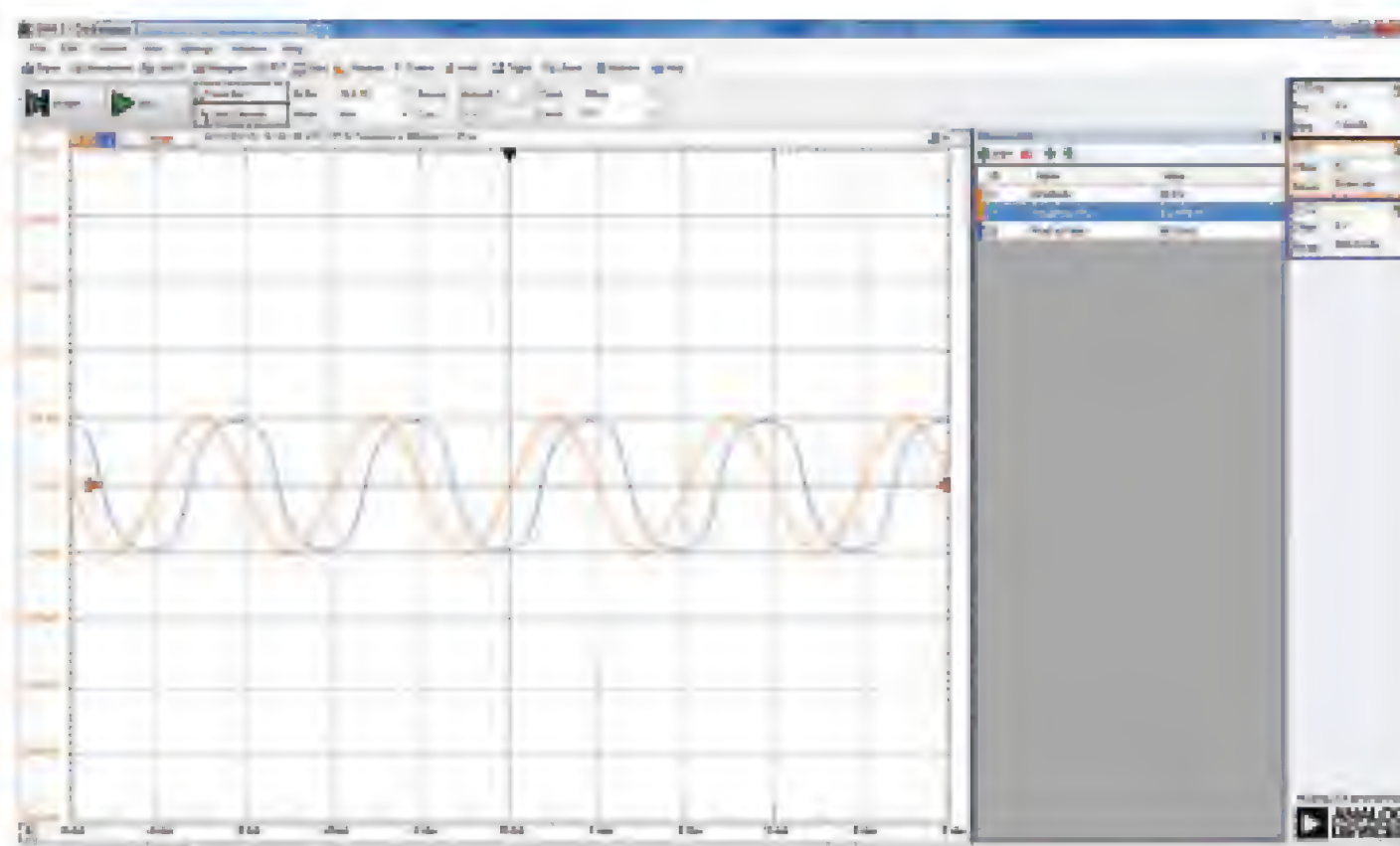
When I first started speaking binary in my pre-microcontroller days, I found that if I needed a manually controlled counter, I had to wire up a counter IC and provide an input clock. After I discovered microcontrollers, the other option — which became the preferred option — was to write a counter program that ran on the target microcontroller.

Digilent, Inc.
Analog Discovery
Discovery BNC
Waveforms
www.digilentinc.com

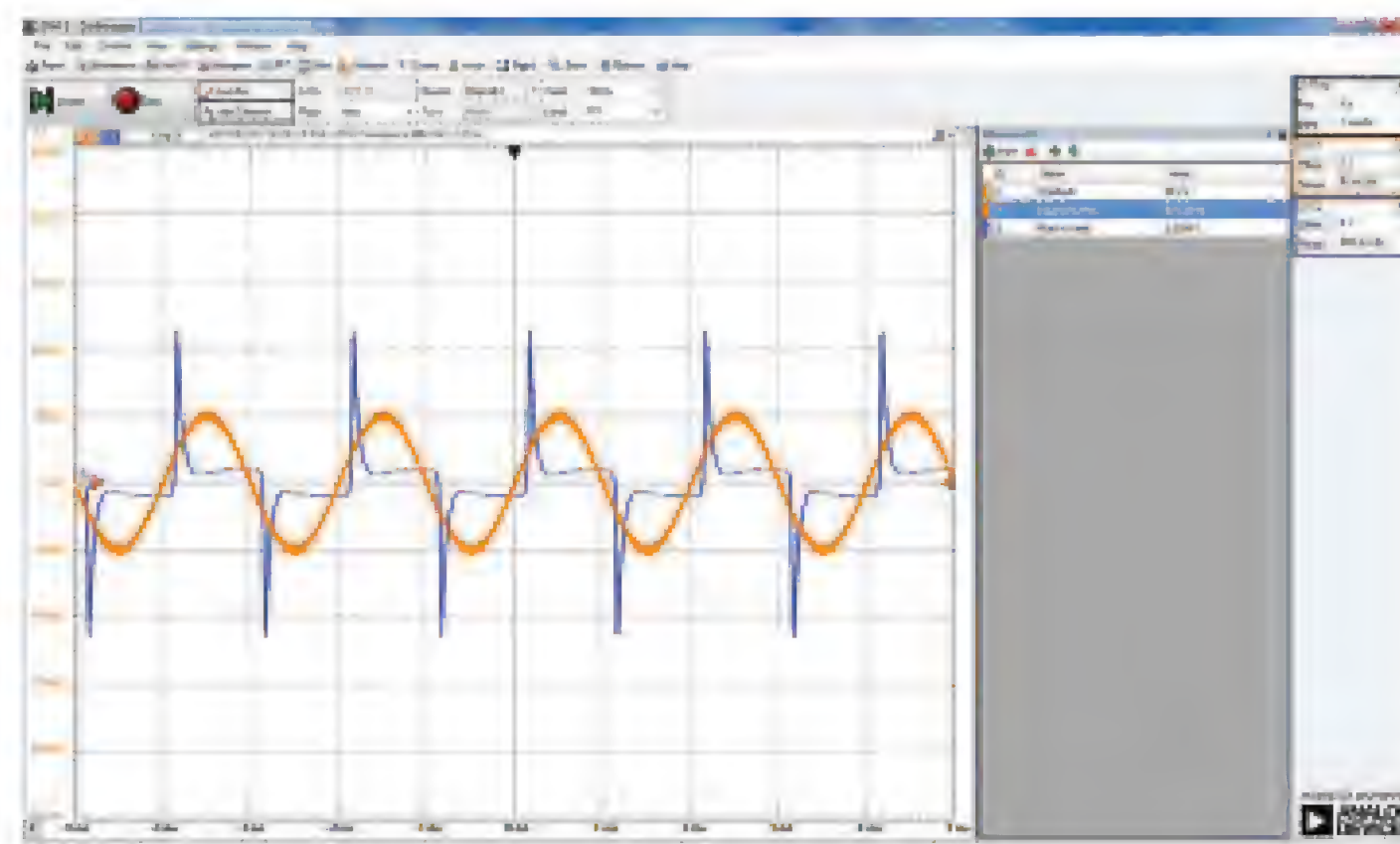
Electro Harmonix
Big Muff Pi
www.electroharmonix.com



■ **Photo 5.** There are two models of the Big Muff Pi. This one is made in New York City. You can also get a version that is made in Sovtek, Russia.

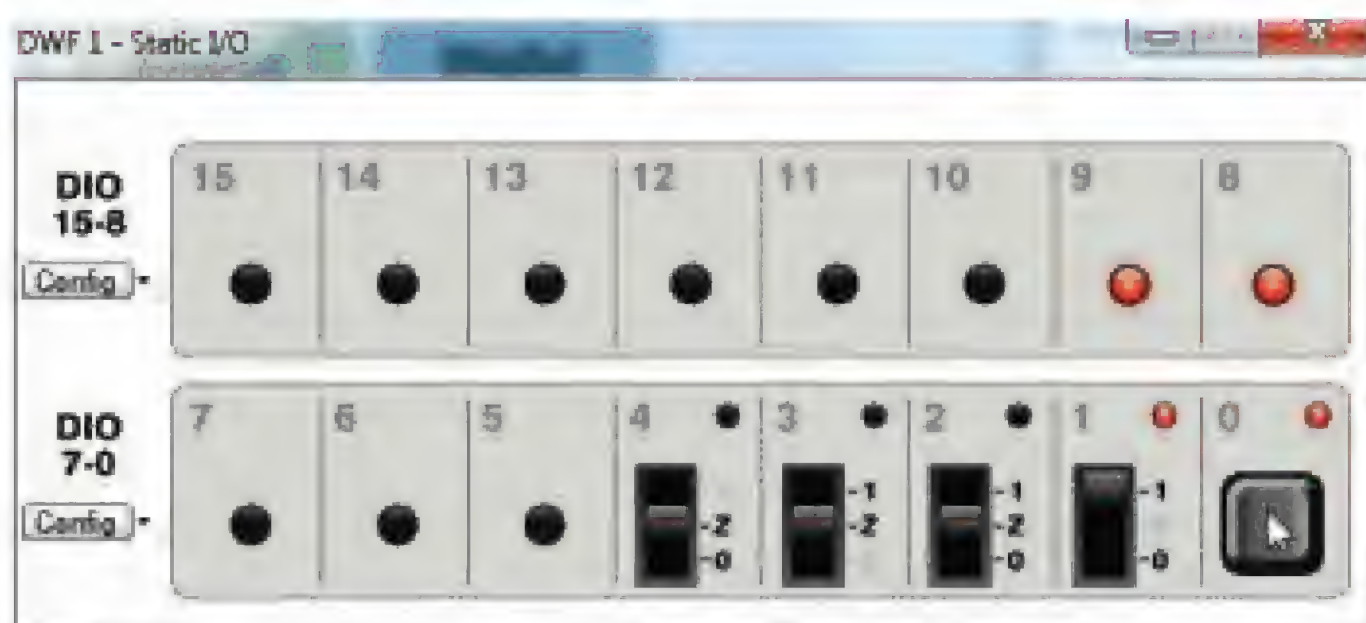


■ **Screenshot 1.** With all of the fuzz control potentiometers at minimum, the Big Muff Pi is still affecting the input signal.

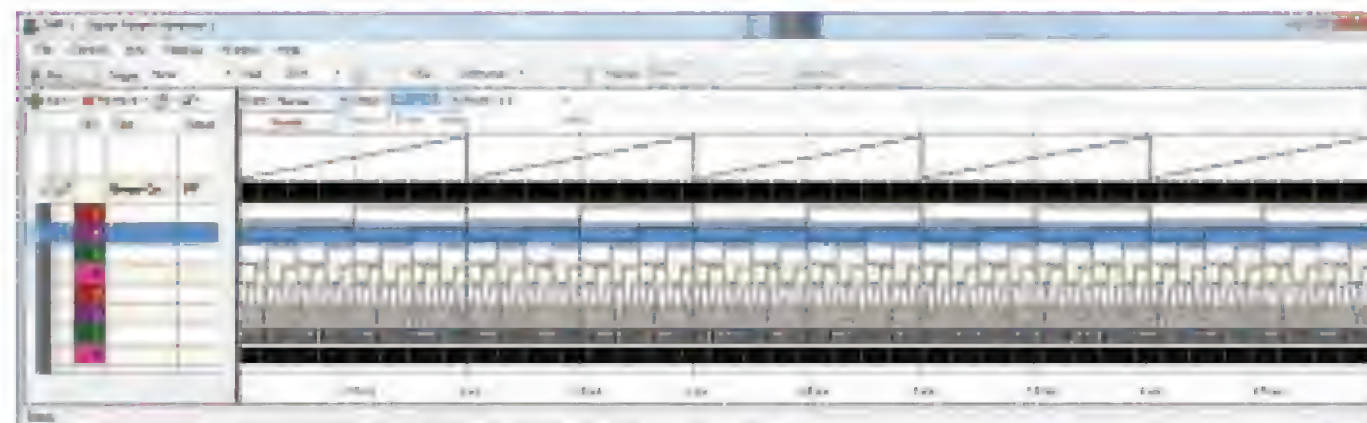


■ **Screenshot 2.** This is what fuzz looks like. I activated the persistence option to emphasize the traces.

The Analog Discovery allows you to simply configure a slider that counts up or down with the click of a mouse button. The Discovery's low byte of GPIO is configured as a slider in **Screenshot 4**. Moving the slider or clicking on the up/down buttons places the resultant binary value on the



■ Screenshot 3. Now, this is the way to add switches and LEDs to a project. The virtual switches and LEDs come in handy when you need to feed a microcontroller input pin or read a microcontroller's output pin state.



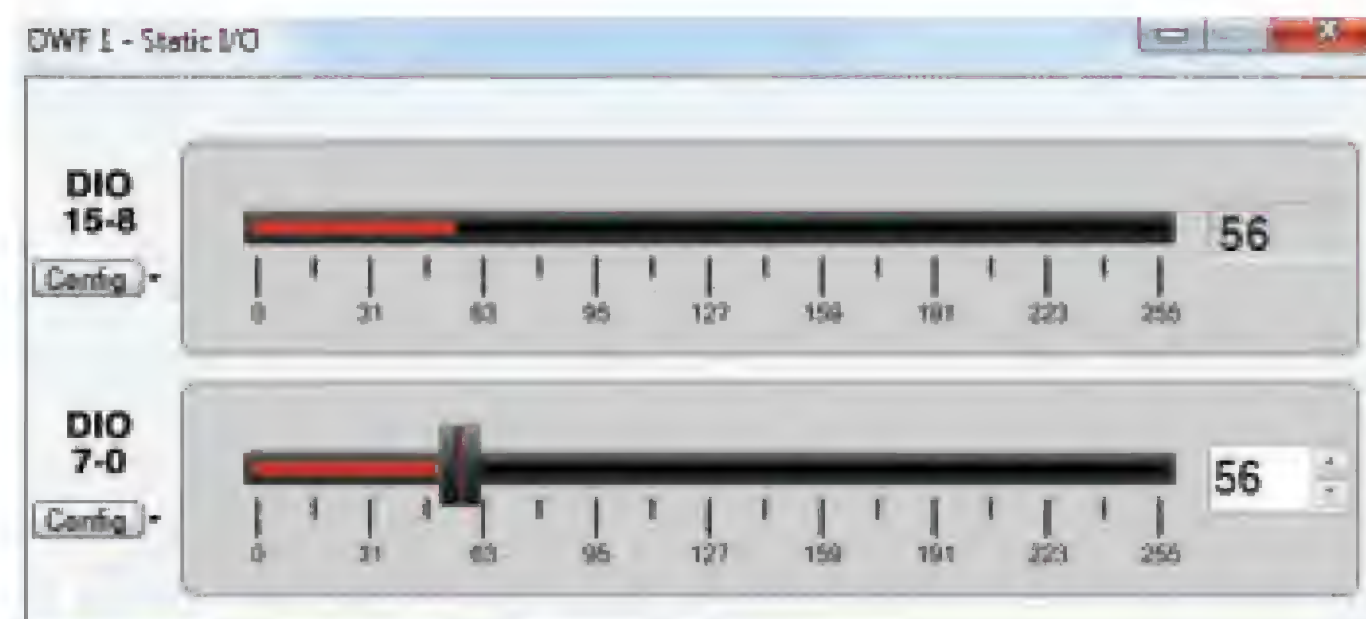
■ Screenshot 7. Microcontroller? We don't need no stinkin' microcontroller. We can manipulate the analog or digital domains by simply fiddling with the Analog Discovery's electronic knobs.



■ Screenshot 4. No code, no compiler, no counter IC, no switches. Just an Analog Discovery creating an instant binary UP/DOWN counter.



■ Screenshot 8. Bits 0 through 7 are the basis for our eight-bit binary counter. I also chose to show the simulated analog value of the bus versus time.



■ Screenshot 5. In this shot, a progress bar replaces the LED display. The lower eight bits of the Analog Discovery GPIO are configured as outputs (slider). The upper eight bits are inputs.



■ Screenshot 6. The pair of sevens is not by design. The seven-segment LED's 8, 9, and 10 bits just happen to be electrically tied to the slider's 0, 1, and 2 bits, respectively.

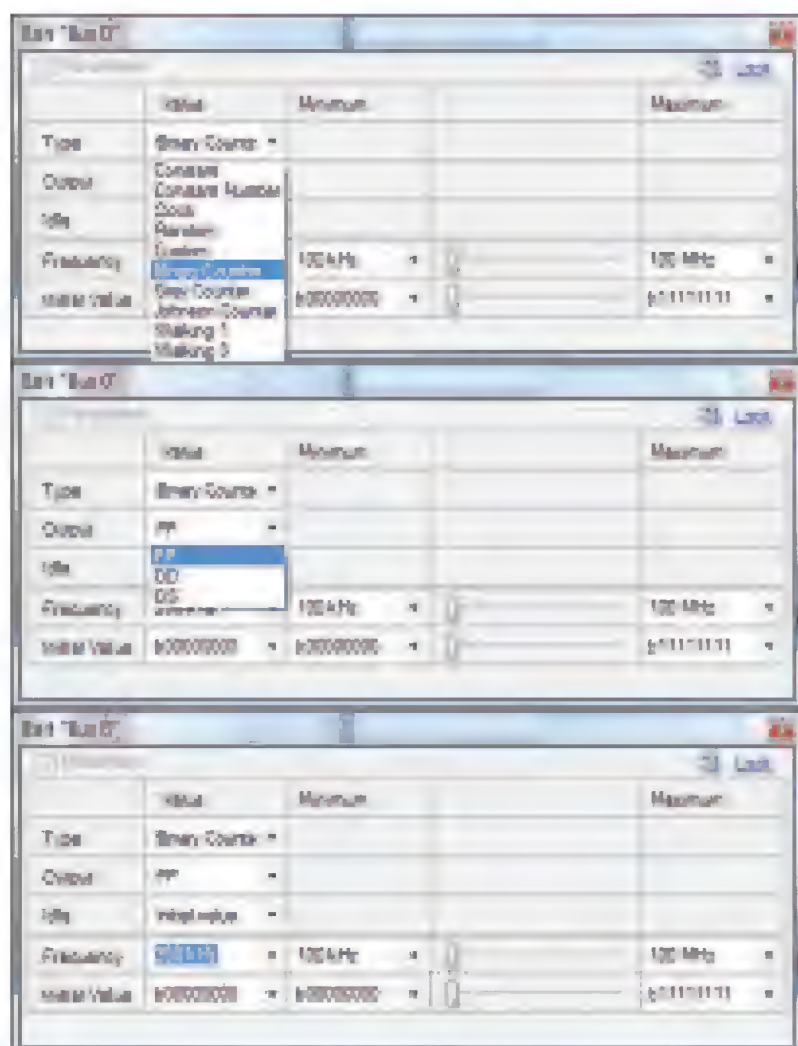
Discovery's low byte of GPIO pins. Again, I used its high byte of GPIO pins as the slider's binary display.

Screenshot 5 retains the slider configuration. However, nothing has really changed logically. The Discovery's high byte of GPIO pins is now representing data on the low byte of GPIO pins as a decimal progress bar value.

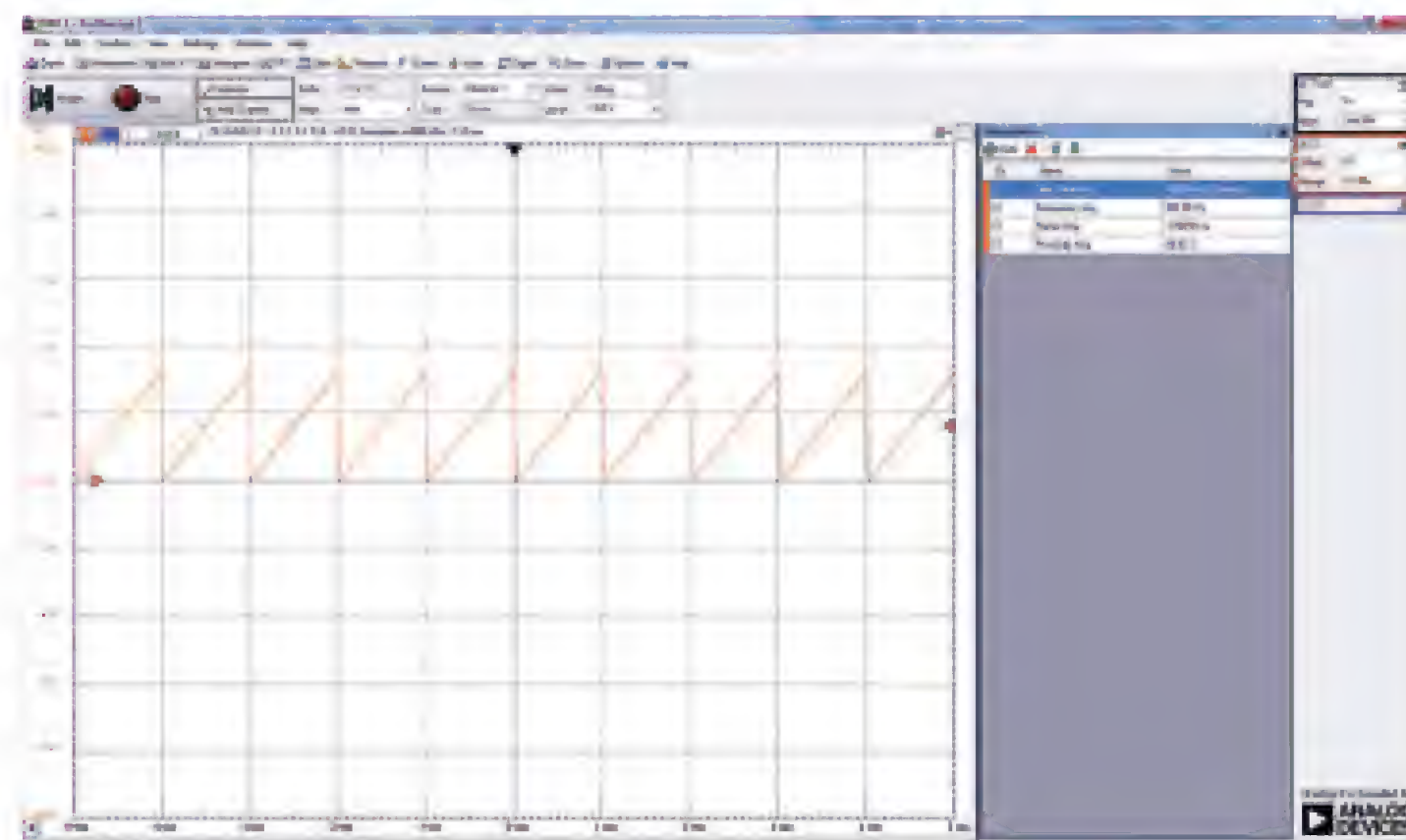
Does 74LS7447 mean anything to you? If it doesn't, the 7447 was the only way to drive a seven-segment LED display module back in the day.

I can still remember when I tossed the 7447 concept and used a PIC and I/O multiplexing to drive a four-digit set of seven-segment display modules. These days, there are many "automated" ways to drive seven-segment LED-based displays.

If you need to emulate a seven-segment LED module, check out **Screenshot 6**. The upper eight bits of the Discovery GPIO pins have now been configured as a seven-segment display. There is no correlation between the slider value and the display LED value. The pair of 7s results from the seven-segment LED's 8, 9, and 10 being attached to the slider's bits 0, 1, and 2.



■ Screenshot 9. You've got to admit, this is a bunch easier than writing code.



■ Screenshot 10. Sixteen precision resistors + one Analog Discovery = 1 kHz sawtooth wave.

Dust in the Wind

I mentioned earlier that the Analog Discovery lets you take the "lab" with you. Well, there's yet another thing you don't have to tote around. That would be a microcontroller. If you stop and think about it, a microcontroller generates wind (digital signals) and collects the resultant dust (analog or digital data).

The Discovery is perfectly capable of generating some wind itself, and that wind is really blowing in **Screenshot 7**. I've opened up the Discovery's digital waveform generator window and dialed in a

Repair Monitors, Power Supplies, Gameboards? Check any Electrolytic Capacitor *In-Circuit* with 100% Accuracy in 3 Seconds---**GUARANTEED***



Automatically discharges capacitor

Checks DCR with alerts for shorts

Measures DCR to 500 ohms

Measures ESR from 0.1 to 20 ohms

Checks caps from .47uF to 2200uF

Beeps one to five beeps for quality

Three-color chart for good-fair-bad

*range 0.47uF - 2.2KuF 90-day money-back guarantee

Portable, Easy to Use, Inexpensive, Highly Recommended by leaders in the industry. **CapAnalyzer 88A** by EDS, Inc.

At your favorite distributor, or call 561-487-6103 www.eds-inc.com



PBP3
PICBASIC PRO™ Compiler

Experimenter - \$49.95
Silver Edition - \$119.95
Gold Edition - \$269.95

The industry-standard BASIC compiler for Microchip PIC® microcontrollers.

Multi-Seat Licensing
for Educational Institutions

Upgrade from
PICBASIC™ Compiler (PBC)

Download a FREE trial version now.

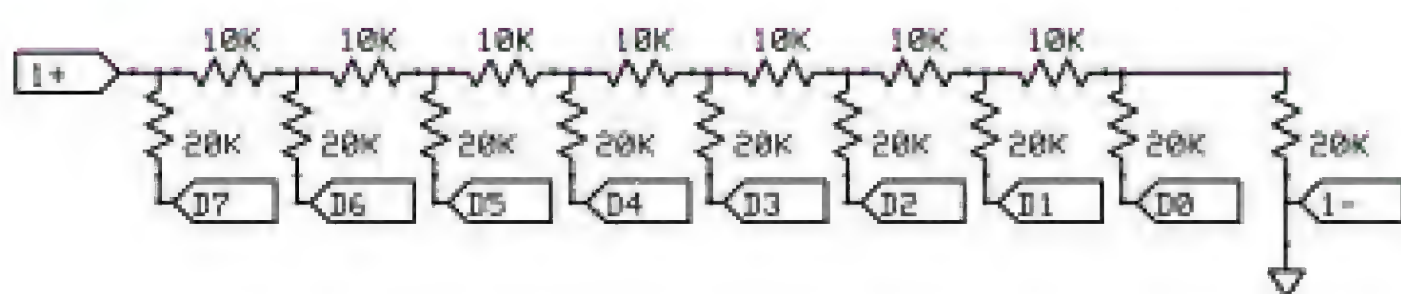
www.PBP3.com

microEngineering Labs, Inc.

www.melabs.com

888-316-1753

PICBASIC and PICBASIC PRO are trademarks of Microchip Technology Inc. in the USA and other countries. PIC is a registered trademark of Microchip Technology Inc. in the USA and other countries.



■ Schematic 1. This is a very crude way to convert digital to analog. However, as you can see in Screenshot 10, it works very well when constructed with 1% resistors.

binary counter on an eight-bit bus.

This whole wind thing began with **Screenshot 8**. I created Bus 0, which consists of bits 0 through 7 of the Discovery's GPIO subsystem. As you can see, there are many data formats available to choose from. This is a no-brainer. We want a binary counter that counts from bit 0 up. So goes our display format selection.

The binary counter option is selected in **Screenshot 9**. Note that we can also create a Gray or Johnson counter.

Here's another advantage of using the Analog Discovery. If you don't have a clue of how a Gray or Johnson counter works, just select it and a logic analyzer preview will appear. The same goes for the walking counters. The output selection is based on the fact that right now nothing is connected to Bus 0. There is no need to select OD (Open Drain) or OS (Open Source), which means we will need to add some resistors to the bus. So, PP (Push Pull) will work just fine. The frequency is arbitrary.

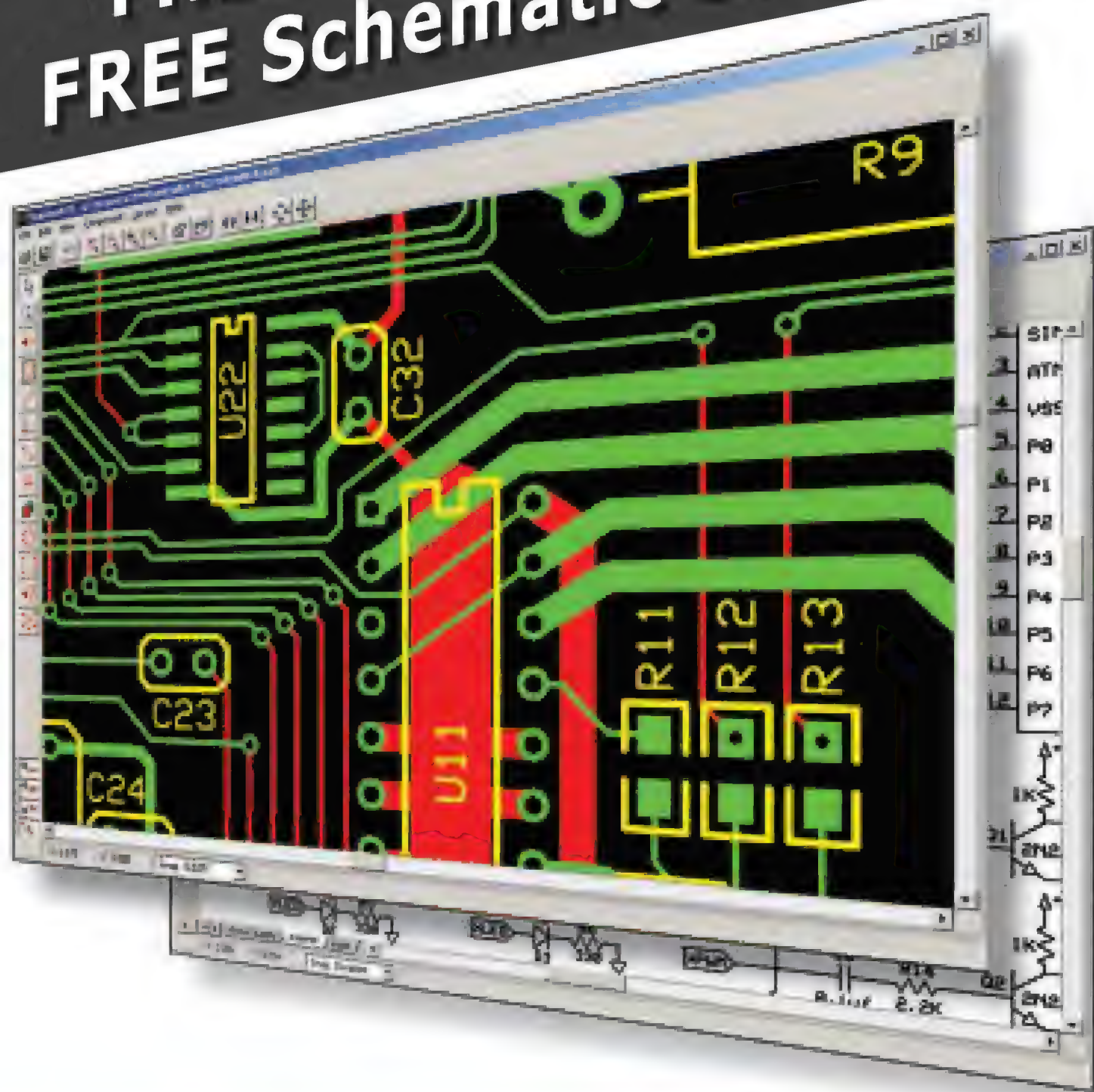
However, choosing 256 kHz means that the complete byte count (0 to 255) will complete every millisecond. You can see this time relationship in the analog emulation view of **Screenshot 7**. Thus, the wind kicked up by the binary counter results in analog dust.

To get a true sample of the dust, we must turn some hardware into the wind. We'll keep it simple and arrange some precision resistors to form an eight-bit DAC. **Schematic 1** is a simple R-2R resistor ladder. I lashed up the R-2R DAC circuit and connected the Analog Discovery to the resistor ladder as shown in the schematic. The oscilloscope image captured in **Screenshot 10** is exactly what we expected. The R-2R resistor ladder excited by our eight-bit counter clocked at 256 kHz results in a 1 kHz sawtooth waveform.

A Book in a Black Box

The Analog Discovery allows you to think it and then immediately do it. Using it is analogous to having an interactive copy of Horowitz and Hill's book, *The Art of Electronics* with you at all times in your design cycle. **NV**

\$51^{For 3} PCBs
FREE Layout Software!
FREE Schematic Software!



- 01 DOWNLOAD our free CAD software
- 02 DESIGN your two or four layer PC board
- 03 SEND us your design with just a click
- 04 RECEIVE top quality boards in just days

expresspcb.com

CLASSIFIEDS

KITS/PLANS

Electronic Kits

- Fun and Educational
- Discount White and Blue LEDs
- Quantity Discounts

www.bakatronics.com

QKITS.COM

We have lots of fun electronic projects for everyone of any skill level!

Stock up now for your winter building!

FREE SHIPPING

On all ground orders over \$175

1-888-GO-4-KITS

MAGNETIC LEVITATION KIT

Float small objects in thin air.
From \$39.75

www.arttec.net/Levitation

COMPONENTS



RF PARTS™ CO.

Tubes, Transistors, Power Components
Email: rfp@rfparts.com • Web: www.rfparts.com

800-737-2787 Fax 888-744-1943

ANTIQUE TUBES OR EQUIV.

Call for free gift!

1L6, 6E5, 50A1, 45, 50, 543, 544, 80, and other antique components

DIALCOVER.COM

SECURITY

SPY EQUIPMENT

BEST PRICES GUARANTEED on CCTV and counter spy products!

914.699.2294

www.mjelectronics.com

SURPLUS

SURPLUS ELECTRONIC PARTS & ACCESSORIES



Over 20,000 Items in Stock.

Cables
Connectors
Displays
Fans

Hardware
LEDs
Motors
Potentiometers

Relays
Semiconductors
Service Manuals
Speakers

Switches
Test Equipment
Tools
VCR Parts

Surplus Material Components
SMC ELECTRONICS
www.smcelectronics.com

No Minimum Order.
Credit Cards and PAYPAL Accepted.
Flat \$4.95 per order USA Shipping.

MISC FOR SALE

HUGE CLOSE-OUT SALE!

All kinds of electronic products and parts!

Test equipment, schematics, tubes, audio-video equipment ... too numerous to list. Call for availability.
loustvelectronics@verizon.net
215.271.1138 or 215.468.3028

WANT TO BUY

WANT TO BUY: WHOLESALE OR MANUFACTURING COMPANY

Inventory, machinery, bank foreclosures, estates. ANYTHING CONSIDERED.

NO COST OR OBLIGATION!

In our 35th year! Visit

www.iseliquidator.com

for our latest list.

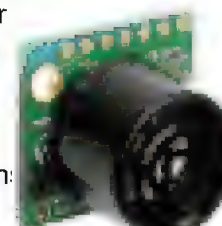
PHONE: 956-444-0004 888-351-5550

ROBOTICS

Ultrasonic Distance Sensing Made EZ

LV-MaxSonar®-EZ™

- Most popular MaxSonar
- Multiple interfaces
- Easy integration
- 1 inch resolution
- Calibrated beam pattern
- Starting at \$29.95



HRXL-MaxSonar®-WR™

- High noise tolerance
- IP67 rated
- 1 mm resolution
- Multi-Sensor operation
- Calibrated beam pattern
- Starting at \$109.95



Check out our live Demo at the

robotics alley

Conference & Expo

Nov. 12-13, 2013 MSP, MN

www.maxbotix.com



www.servo-magazine.com

ENERGY SOLUTIONS

Save Money On Your Electric Bill

www.sunny-solutions-shop.com

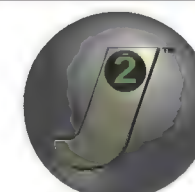
800-315-0309

LIGHTING

Reactive Lighting

Automatic Stair Lighting

www.ReactiveLighting.com



- ◇LED modules
- ◇Solar controls
- ◇Dusk to dawn
- ◇Dimmers 12 V

eBay store name:

J2 LED LIGHTING LLC

<http://stores.ebay.com/j2ledlightingllc>

BOOKS

Free AI Book and Software.

IBM PC.

Write:

davethecoyote@hotmail.com

THE OWNERSHIP, MANAGEMENT, AND CIRCULATION STATEMENT OF NUTS & VOLTS MAGAZINE, Publication Number: 1528-9885 is published monthly. Subscription price is \$26.95. 7. The complete mailing address of known office of Publication is T&L Publications, Inc., 430 Princland Ct., Corona, Riverside County, CA 92879-1300. Contact Person: Larry Lemieux. Telephone: (951) 371-8497. 8. Complete Mailing address of Headquarters or General Business Office of Publisher is T&L Publications, Inc., 430 Princland Ct., Corona, CA 92879. 9. The names and addresses of the Publisher, and Associate Publisher are: Publisher, Larry Lemieux, 430 Princland Ct., Corona, CA. 92879; Associate Publisher, Robin Lemieux, 430 Princland Ct., Corona, CA 92879. 10. The names and addresses of stockholders holding one percent or more of the total amount of stock are: John Lemieux, 430 Princland Ct., Corona, CA 92879; Lawrence Lemieux, 430 Princland Ct., Corona, CA 92879; Audrey Lemieux, 430 Princland Ct., Corona, CA 92879. 11. Known Bondholders, Mortgagees, and other security holders: None. 12. Tax Status: Has not changed during preceding 12 months. 13. Publication Title: Nuts and Volts 14. Issue Date for Circulation Data: October 2012-September 2013. 15. The average number of copies of each issue during the proceeding twelve months is: A) Total number of copies printed (net press run); 33,798 B) Paid/Requested Circulation (1) Mailed Outside County subscriptions: 23,130 (2) Mailed In-County subscriptions: 0 (3) Paid Distribution Outside the Mail including Sales through dealers and carriers, street vendor, and counter sales and other paid distribution outside USPS: 7,309 (4) Paid Distribution by other classes of mail through the USPS: 0; C) Total Paid Distribution: 30,439; D) Free or Nominal Rate Distribution by mail and outside the mail (1) Free or Nominal Rate Outside-County Copies: 0 (2) Free or Nominal Rate In-County Copies: 0 (3) Free or Nominal Rate Copies Mailed at other classes through the USPS: 0 (4) Free or Nominal Rate Distribution Outside the mail: 367; E) Total Free or Nominal Rate Distribution: 367; F) Total Distribution: 30,805; G) Copies not distributed: H) Total: 2,992; Percent paid circulation: 98.81%. Actual number of copies of the single issue published nearest the filing date is September 2013; A) Total number of copies printed (net press run) 32,333; B) Paid/Requested Circulation (1) Mailed Outside County subscriptions: 22,491 (2) Mailed In-County subscriptions: 0 (3) Paid Distribution Outside the Mail including Sales through dealers and carriers, street vendor, and counter sales and other paid distribution outside USPS: 6,656 (4) Paid Distribution by other classes of mail through the USPS: 0; C) Total Paid Distribution: 29,047; D) Free or Nominal Rate Distribution by mail and outside the mail (1) Free or Nominal Rate Outside-County Copies: 0 (2) Free or Nominal Rate In-County Copies: 0 (3) Free or Nominal Rate Copies Mailed at other classes through the USPS: 0 (4) Free or Nominal Rate Distribution Outside the mail: 100; E) Total Free or Nominal Rate Distribution: 100; F) Total Distribution: 29,147; G) Copies not distributed: 3,186; H) Total: 32,333; Percent paid circulation: 99.66%. I certify that these statements are correct and complete. Lawrence Lemieux, Publisher - 10/07/2013.

ES Excess Solutions
156 S. Milpitas Blvd.
Milpitas, CA 95035

Come in and see our retail store!

We Buy & Sell Excess Electronic Inventory

(408) 262-3900

www.excesssolutions.com

HARDWARE WANTED

DEC EQUIPMENT WANTED!!!

Digital Equipment Corp. and compatibles.
Buy - Sell - Trade

CALL KEYWAYS 937-847-2300
or email buyer@keyways.com

Subscribe today!
www.nutsvolts.com

>>> QUESTIONS

Tesla Coil Theory

How does a Tesla coil actually work? I'm especially interested in the relationship between the primary winding and secondary winding that creates the spark.

#11131

Brian Miller
St. Paul, MN

Motor Selection

I want to rotate one 40 lb metal triangle up 90°. Its base will be connected by a hinge to the base of another triangle. The base is 67.88"; the other legs being 48", and a height of 57.87". By rotating the apex of the triangle 90°, the apex would travel an arc by 8.18 ft. I intend to use two sprockets and a chain connected to a motor. What size motor will I need to use to complete this and how is it determined? I understand $w = fd$ and $P = fd/t$. I would like the time to be six seconds.

Can someone please help with the answer and also show me how to figure this problem out for myself?

#11132

Derek
Oklahoma

Filter Caps and Power Supplies

This has to do with electrolytic aluminum filter caps for switching power supplies.

No matter what type filter cap I try, they blow out (become pregnant) after months or a few years. I repaired cable boxes for many years that had the exact same problem.

This is only a three volt supply at about two amps. Ten volt 1,000 mfd

caps are used in the stock supply. Also, a three amp Schottky diode (burns up) supplies the DC to a 15 amp logic N-channel MOSFET with a heatsink. It gets hot. Then, the output of it gets a cap, a choke, and a cap. Nicely filtered three volts.

This is my final change-out and it is lasting the longest. So far, no blow outs, but it has only been seven months.

Now, the two five amp Schottky diodes in parallel. Using only one still gets super hot. Caps 25 volt at 1,000 mfd. I'm only using general type filter caps at 20% 105° C. Why has this been such a big problem?

The cap that usually blows is the first one after the MOSFET. I see no spikes on the output of the MOSFET either. I could use a TO-220 pack with dual diodes in it, but no room. The two 5 amp in parallel work just great and only get warm.

#11133

Dan Zielinski
Port Saint Lucie, FL

>>> ANSWERS

[#9131 - September 2013]

Lead-Free Soldering

I've finally used up my 2 lb roll of Kester lead-based solder and I'm ready to move to lead-free soldering. Is there anything I need to do to my existing soldering equipment to make the move? Anything else I should know?

Making the move to leadless soldering is a great idea, given the health concerns of lead toxicity —

especially if you have children in the house or workshop. It's important to note that the rosin used as a flux in solder can be just as problematic as the lead. You can end up with an asthma-like condition if you don't work in a well-ventilated area.

Richard Pena
via email

[#10136 - October 2013]

Battery Disposal

I have a box full of old lithium batteries. I know they're not supposed to be put in the normal trash, but I don't want to pay a fee to have 'hazardous waste' removed. Is there a safe, environmental friendly way for me to dispose of the batteries that won't cost me time and money?

#1 Stores such as Walmart usually have "fishbowls" at the jewelry counter for depositing coin batteries. I would imagine that jewelry stores and hearing aid centers would also have disposal options that may not be limited to themselves or their customers only.

Worse than coin batteries — which are used to a lesser extent — D, C, 9V, and especially AA and AAA batteries are a bigger problem for disposal. These days, manufacturers are using batteries for power far more than in years past, finding that UL approval is much easier under battery power. And if it's electronic, it has a remote or something wireless and the larger batteries abound.

Some cities have recycling centers. Again, Walmart may have a recycling plan. There are stores that

WHATSOEVER are made by the publisher. The implementation of any answer printed in this column may require varying degrees of technical experience and should only be attempted by qualified individuals.

Always use common sense and good judgment!

Send all questions and answers by email to forum@nutsvolts.com or via the online form at www.nutsvolts.com/tech-forum

All questions AND answers are submitted by *Nuts & Volts* readers and are intended to promote the exchange of ideas and provide assistance for solving technical problems. All

submissions are subject to editing and will be published on a space available basis if deemed suitable by the publisher. Answers are submitted by readers and **NO GUARANTEES**

Send all questions and answers by email to forum@nutsvolts.com
or via the online form at www.nutsvolts.com/tech-forum

deal only in batteries and they may be able to recycle. Many "big box" stores have recycling available for rechargeable batteries. You just have to ask at places dealing with products that are prone to battery power.

Dean Huster
via email

#2 Visit your local Lowe's store. The ones in my area of the East Coast have a recycling bin near the returns desk that takes CFLs and rechargeable batteries. No muss or fuss, just drop them in!

Home Depot also takes them at the customer service desk. I just box them and mark the box 'Battery Recycling' and there has been no problem.

Len Powell
Finksburg, MD

[#10134 - October 2013]

Can Resistors Make the Difference?

I'm working on an audiophile-quality all-transistor amp. I was told that carbon film resistors are less noisy than wire-wound precision resistors — which are more expensive. Could I hear the difference in resistors? Is there a theoretical noise difference?

Carbon composition resistors are THE WORST THINGS TO USE in an audiophile amp as their "absolute value" tolerances range from 10-20%. Plus, their "absolute resistance" values are very susceptible to heating and cooling effects (i.e., they get warm and their resistance changes — sometimes dramatically!).

Instead of using wire-wounds, use

metal-film resistors. Their tolerances are 5% or less, they're very stable overall, and aren't as susceptible to temperature effects like carbons. Plus, they're much cheaper than wire-wounds.


Ken Simmons
Auburn, WA

[#10135 - October 2013]

Audio Amp ICs

I need a good, single-source audio amp IC for a project. The LM-386 isn't powerful enough. Is there a 'super' LM-386 op-amp that's also single-source (only one power connection)?

Depending on your power and voltage requirements, I would suggest the LM380 or LM384. I have used


THE ORIGINAL SINCE 1994
PCB-POOL
Beta LAYOUT

PCB Prototypes & small series

 **FREE SMT Stencil**
with every prototype order

 **Embedded RFID**
authenticate, track & protect
your product
NEW!
www.magic-pcb.com

Call Tony: 1 707 447 7744
sales@pcb-pool.com

PCB-POOL® is a registered trademark of **Beta**
LAYOUT create electronics

www.pcb-pool.com

Save Up To 60% On Electronic Components

 **Ethernet Mini Board**
*Includes W5100 Embedded TCP/IP IC
*Works with Most Network Protocols
*3.3V or 5V Board Operation
*Ideal for Adding Network Connectivity to your Project
Only \$19.90

 **New Marine Solar Charge Controller**
*Especially Designed for Marine Use
*Suitable for Both 12V and 24V Systems
*Waterproof and Corrosion Resistant
Only \$39.90

 **New microSD Mini Board**
*Ideal for Saving and Recording Real-Time Readings
*Works with both 3.3V and 5V Microcontrollers
*Standard Communications Interface
Only \$9.40

We are your one-stop shop for Microcontroller Boards, PCB Manufacture and Electronic Components

www.futurlec.com

Send all questions and answers by email to forum@nutsvolts.com
or via the online form at www.nutsvolts.com/tech-forum

both of these parts many times over the years. The LM380 is available in an eight-pin or 14-pin package. Unless you are trying to get more than one watt out of the part, you probably won't need a heatsink. Just put a large copper area on your PCB for pins 3, 4, 5, 10, 11, and 12. Both parts are available from Mouser or Digi-Key.

Ray Buck
Phoenix, AZ

[#10138 - October 2013]
Surplus Parts for Amps

I'm just getting into guitar amplifiers, and I'm having a hard time

finding parts — tubes, transformers, and high-voltage capacitors that are affordable. What are the best surplus sources, short of taking apart old amps?

Antique Radio Supply at www.tubesandmore.com is a good source for tube-era components. Right here in the pages of N&V, Canadians David and Babylyn Cantelon have advertised tubes, capacitors, and resistors for tube-era electronics and are at www.justradios.com.

Dean Huster
via email

[#101311 - October 2013]
Save a Light

I'm constantly replacing flashlights because of swollen D cells that expand and get stuck in the flashlight barrel. Is there some way to remove the cells without ruining the flashlight or splattering battery gunk all over?

Do an Internet search under "repairing Maglites." You'll find several sites with instructions and/or videos to help you in this endeavor. Look at them all before proceeding.

Dean Huster
via email

■ LOOK FOR ■ SEARCH FOR ■ FIND
Find your favorite advertisers here!

ADvertiser INDEX

AMATEUR RADIO AND TV

Ramsey Electronics82-83

BATTERIES/CHARGERS

Cunard Associates27

BUYING ELECTRONIC SURPLUS

Jaycar Electronics61
Weirdstuff Warehouse27

CAD/EDA TOOLS

Labcenter Electronics...Back
Cover

CCD CAMERAS/VIDEO

Ramsey Electronics82-83

CIRCUIT BOARDS

AP Circuits59
Cunard Associates27
Dimension Engineering.....47
ExpressPCB76
Front Panel Express LLC39
Futurlec79
GHI Electronics5
PCB Pool79
Saelig Company Inc.59

CNC ROUTER

Probotix7

COMPONENTS

Can Kit Corp.52
Electronix Express45
Futurlec79
SDP/SI27

COMPUTER

Hardware
Weirdstuff Warehouse27

Microcontrollers / I/O Boards

Abacom Technologies23
C2 Robotics.17
GHI Electronics5
Images Co.39
microEngineering Labs75
MikroElektronika3
Pololu Robotics & Electronics19

Software

Labcenter Electronics...Back
Cover

DESIGN/ENGINEERING/ REPAIR SERVICES

Can Kit Corp.52
ExpressPCB76
Front Panel Express LLC39
Images Co.39
PCB Pool79

DRIVE COMPONENT CATALOGS

SDP/SI27

EDUCATION

Command Productions25
NKC Electronics27
Poscope60

EMBEDDED TOOLS

NetBurner2

ENCLOSURES

Front Panel Express LLC39

EVENTS

Autonomous Snowplow Comp. 65
CES81

KITS & PLANS

Can Kit Corp.52
Jaycar Electronics61
NetBurner2
NKC Electronics27
QKITS27
Ramsey Electronics82-83

LINEAR ACTUATORS

Firgelli60

MISC./SURPLUS

All Electronics Corp.58
Front Panel Express LLC39
Scientifics33
Weirdstuff Warehouse27

MOBILE RADIO MOUNTS

E Tip Inc.27

MOTORS

Firgelli60

PROGRAMMERS

C2 Robotics.17
Futurlec79
microEngineering Labs75
MikroElektronika3

RF TRANSMITTERS/ RECEIVERS

Abacom Technologies23
E Tip Inc.27

ROBOTICS

Firgelli60
GHI Electronics5
Lemos International Co.27
Pololu Robotics & Electronics.....19
Probotix7
Robot Power27
SDP/SI27

SATELLITE

Lemos International Co.27

SECURITY

E Tip Inc.27

SOLDERING STATIONS AND IRONS

HAKKO24

TEST EQUIPMENT

Dimension Engineering.....47
Electronic Design Specialists75
Images Co.39
Jaycar Electronics61
NKC Electronics27
Poscope60
Saelig Company Inc.59

TOOLS

C2 Robotics.17
HAKKO24
MikroElektronika3
NetBurner2
PanaVise32
Poscope60

Abacom Technologies23
All Electronics Corp.58
AP Circuits59
Autonomous Snowplow
Competition65
Can Kit Corp.52
C2 Robotics.17
CES81
Command Productions25
Cunard Associates27
Dimension Engineering.....47
Electronic Design Specialists 75
Electronix Express45
E Tip Inc.27
ExpressPCB76
Firgelli60
Front Panel Express LLC ...39
Futurlec79
GHI Electronics5
HAKKO24
Images Co.39
Jaycar Electronics.....61
Labcenter
ElectronicsBack Cover
Lemos International Co.27
microEngineering Labs.....75
MikroElektronika3
NetBurner2
NKC Electronics27
PanaVise32
PCB Pool79
Pololu Robotics & Electronics 19
Poscope60
Probotix7
QKITS27
Ramsey Electronics82-83
Robot Power27
Saelig Company Inc.59
Scientifics33
SDP/SI27
Weirdstuff Warehouse27

YOU CAN'T ALWAYS PREDICT THE FUTURE.

Tuesday, January 7
through Friday, January 10, 2014
Las Vegas, Nevada • CESweb.org • #CES2014

**BUT THESE FOUR
DAYS CAN GIVE YOUR
EDUCATED GUESS A Ph.D.**

Over four days, those who shape the future gather in Las Vegas. Here, brands, markets and economies converge in what's far more than a tradeshow. And in 2014, there's more opportunity than ever to connect with those who matter. All that's missing now is you. **Register today at CESweb.org.**



THE GLOBAL STAGE FOR INNOVATION

PRODUCED BY  CEA®

Get Ready For The Holidays With a Ramsey Kit!

Digital Controlled FM Stereo Transmitters

- ✓ PLL synthesized for drift free operation
- ✓ Front panel digital control and display of all settings and parameters!
- ✓ Professional metal case for noise-free operation
- ✓ EMI filtering on audio and power inputs
- ✓ Super audio quality, rivals commercial broadcasts
- ✓ Available in domestic kit or factory assembled export versions

For more than a decade we've been the leader in hobbyist FM radio transmitters. We told our engineers we wanted a new technology transmitter that would provide FM100 series quality without the advanced mixer features. They took it as a challenge and designed not one, but TWO transmitters!



The FM30 is designed using through-hole technology and components and is available only as a do-it-yourself kit with a 25mW output very similar to our FM25 series. Then the engineers redesigned their brand-new design using surface mount technology (SMT) for a very special factory assembled and tested FM35WT version with 1W output for our export only market!

All settings can be changed without taking the cover off! Enter the setup mode from the front panel and step through the menu to make all of your adjustments. A two line LCD display shows you all the settings! In addition to the LCD display, a front panel LED indicates PLL lock so you know you are transmitting.

Besides frequency selection, front panel control and display gives you 256 steps of audio volume (left and right combined) as well as RF output power. A separate balance setting compensates for left/right differences in audio level. In addition to settings, the LCD display shows you "Quality of Signal" to help you set your levels for optimum sound quality. And of course, all settings are stored in non-volatile memory for future use!

Both the FM30 and FM35WT operate on 13.8 to 16VDC and include a 15VDC plug-in power supply. The stylish black metal case measures 5.55"W x 6.45"D x 1.5"H. (Note: After assembly of this do-it-yourself hobby kit, the user is responsible for complying with all FCC rules & regulations within the US, or any regulations of their respective governing body. FM35BWT is for export use and can only be shipped to locations outside the continental US or valid APO/FPO addresses or valid customs brokers for end delivery outside the continental US.)

FM30B Digital FM Stereo Transmitter Kit, 0-25mW, Black \$199.95
FM35BWT Digital FM Stereo Transmitter, Assembled, 0-1W, Black (Export ONLY) \$299.95

Electrocardiogram ECG Heart Monitor

- ✓ Visible and audible display of your heart rhythm!
- ✓ Bright LED "Beat" indicator for easy viewing!
- ✓ Re-usable hospital grade sensors included!
- ✓ Monitor output for professional scope display
- ✓ Simple and safe 9V battery operation

Use the ECG1C to astound your physician with your knowledge of ECG/EKG systems. Enjoy learning about the inner workings of the heart while, at the same time, covering the stage-by-stage electronic circuit theory used in the kit to monitor it. The documentation with the ECG1C covers everything from the circuit description of the kit to the circuit description of the heart! Multiple "beat" indicators include a bright front panel LED that flashes with the actions of the heart along with an adjustable level audio speaker output that supports both mono and stereo hook-ups. In addition, a monitor output is provided to connect to any standard oscilloscope to view the traditional style ECG/EKG waveforms just like you see on ER... or in the ER! 10 hospital grade re-usable probe patches are included together with the matching custom case set shown. Safe 9V battery operation.

ECG1C Electrocardiogram Heart Monitor Kit With Case & Patches \$44.95
ECG1WT Electrocardiogram Heart Monitor, Factory Assembled & Tested \$89.95
ECGP10 Electrocardiogram Re-Usable Probe Patches, 10-Pack \$4.95

HV Plasma Generator

Generate 2" sparks to a handheld screwdriver! Light fluorescent tubes without wires! This plasma generator creates up to 25kV at 20kHz from a solid state circuit! Build plasma bulbs from regular bulbs and more! Runs on 16VAC or 5-24VDC.

PG13 HV Plasma Generator Kit \$64.95

Van de Graaff Generators

Create your own lightning with these time tested student favorites! Produces low current static currents that can be "shocking" but perfectly safe! Draw sparks to a screwdriver, grab hold and watch your hair stand straight up! Two models produce from 200kV to 400kV!

VG Van de Graaff Generators from \$139.95

Retro Nixie Tube Clocks

Genuine Nixie tubes popular in the 50's now brought back in the neatest digital clocks around today!

Enjoy yesterday's high tech marvels today with our complete line of Nixie Tube Clocks! 6-digit hand-crafted teak & maple hardwood base, 12/24 hour format, soft fade-out, auto-dim, crystal time base, date, DST, AM/PM, programmable LED mood colors, optional GPS time reference, and a whole lot more! Or a mini 4-digit version in a similar hand rubbed base. Also available in a modern polished aluminum base or a futuristic clear acrylic see-through base! Visit our web site for details and specs on all models and the GPS clock reference.

NIXIE Nixie Tube Clock Kits from \$229.95

The Holiday Spotlight!

LED Animated Santa

This animated Santa and reindeer display has been our most popular holiday display for years! It contains a whopping 126 dazzling colored LEDs which make it a great holiday sign that is guaranteed to draw attention!

LED animated motion makes it come alive. Runs on standard 9V battery or 9-12VDC external power supply. Dazzle your friends this great display!

MK116 LED Animated Santa Kit \$21.95

LED Christmas Tree

Electronic Christmas tree features 134 bright colored LEDs in the shape of a gorgeous holiday Christmas tree. Includes 18 random flashing blinking "candles" on the PC board! Runs on a 9V battery or external 9-12VDC power supply.

MK117 LED Christmas Tree Kit \$24.95

LED Animated Holiday Bell

This PC board holiday bell is animated to simulate a bell swinging back and forth! 84 bright colored LEDs will dazzle you with holiday cheer! Includes an on/off switch. Runs on 9V.

MK122 LED Animated Bell Kit \$18.95

3D LED Christmas Tree

Not your average LED display! 4 branch sections give this tree a true 3D look! 16 red LEDs light it up with yellow LEDs for you to customize your tree! The base of the tree is actually the 9V battery acting as a self supporting base! Now that's pretty neat!

MK130 3D LED Christmas Tree Kit \$7.95

SMT LED Christmas Tree

Build this subminiature Christmas tree and learn SMT at the same time. Small enough to wear as a badge or pendant! Extra SMT parts are included so you can't go wrong! Runs on Li-Ion cell.

MK142 SMT LED Christmas Tree Kit \$10.95

SMT LED Smiley Face

This is a great attention grabber and also teaches you the basics of SMT construction! Perfect to wear through the holiday season or to hang on your tree as an attention getting ornament! Extra parts included! Runs on Li-Ion cell.

MK141 SMT LED Smiley Kit \$9.95

LED Traffic Signal

Not exactly a holiday theme, a real attention getter for this time of the season! Impress your friends with this neat 4-way traffic signal! Operates just like a standard signal, and features adjustable delay. Red, yellow, and green LEDs are used just like the real thing! Runs on 9V battery.

MK131 LED Traffic Signal Kit \$7.95

LED Switcher Blinky

Wait, an LED that runs on 3VDC running on 1.5VDC? Learn power supply switching and end up with a super bright Telux LED blinking at 140 kHz! Great to light up your ornaments! Runs on a single standard AA battery (not included).

LSW1 LED Switcher Blinky Kit \$14.95



Where Electronics...
Is Always Fun!

- ✓ Build It!
- ✓ Learn It!
- ✓ Achieve It!
- ✓ Enjoy It!

Follow Us and SAVE \$\$

Follow us on your favorite network site and look for a lot of super deals posted frequently... exclusively for our followers!



Stereo Audio Gain Controller

- ✓ Stereo audio processing while preserving audio dynamics!
- ✓ True stereo control keeps virtual sonic source location intact!
- ✓ Auto-bypass restores original levels when power is turned off!
- ✓ Built-in bar graph indication of signal level with display mute!

The SGC1 is one of our latest kits, and provides a great solution to the age-old problem: how can we easily correct inconsistent audio levels without negatively affecting the dynamics of the audio signal? The SGC1 circuit implements a principle known as the "Platform Gain Principle," which was originally developed by CBS Labs (what we now know as CBS in the TV and radio world) to allow transmitted audio levels to be automatically adjusted to keep them within a desired range.

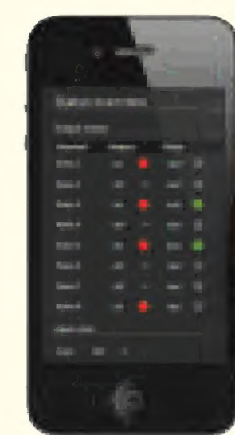
Think of it like an audio engineer, constantly adjusting the output level in order to limit highs that would be too loud while boosting lower levels so that they can still be heard. You may think "oh, this is just another limiter/compressor!" Not so! Here's the real trick: keeping the full dynamic range ratio of the output signal the same as the original input - something the typical limiter/compressor can only dream of doing! The SGC1 can be placed in just about any standard analog stereo line level audio circuit (the red and white RCA connectors or the mini-phone connector) to keep the audio level within the desired range. It's also the perfect addition to any of our hobby kit transmitters, allowing you to match levels between different audio sources while keeping lows audible and preventing the highs from overdriving. The SGC1 makes a great addition to any audio system where you need to keep levels from different sources under control, but still make sure they all sound great! In addition to its useful basic function and great audio performance, the SGC1 also boasts a front panel LED meter to give an indication of the relative level of the input signal, plus a level control (also on the front panel) that allows you to adjust the controller to the min/max center point of your desired level range. And yes, it is a **Stereo Gain Controller!** Meaning that the levels of both the left and right channels are monitored and adjusted equally, thereby maintaining the relative virtual position of things like instruments, singers and speakers! The entire unit is housed in a slim attractive black textured aluminum case that is sure to complement your studio or home theatre. If you're looking for perfect audio levels, hire a broadcast audio engineer, but if that doesn't fit your budget, the SGC1 is the next best thing! Includes 15VDC world-wide power adapter.

SGC1 Stereo Audio Platform Gain Controller Kit

\$179.95

8-Channel Remote Ethernet Controller

Now you can easily control and monitor up to 8 separate circuits via the standard Ethernet network in your home or office. Connection wise it couldn't be simpler. The controller functions as an IP based web server, so it can be controlled by any internet browser that can reach your network! There are no drivers or proprietary software required, just access the controller like any web page from your PC, laptop, or even your smartphone!



Security is assured allowing up to 4 separate user credentials. The controller can be set to a specific static IP within your network subnet or can be set to DHCP (auto negotiate). The controller can even be programmed to send you an email to notify and confirm power up and status changes!

To simplify the connection of your equipment to the controller, 8 separate and isolated relay outputs are provided! This gives you internet or network control of up to 8 separate functions. No need to worry about interfacing a logic high or logic low, or burning up the interface! The applications are endless! From something as simple as turning on and monitoring lights at your house with a normal latched closure to advanced control of your electronic gadgets, radio equipment, or even your garage door!



Each relay contact is rated at 12A at 30VDC or 16A at 230VAC. Each of the 8 channels has built-in timer and scheduler programs for day, weekend, working days, every day, and every day except Sunday. Relay control functions are programmable for on, off, or pulse (1-99 seconds, 1-99 minutes, or 1-99 hours). In addition to control functions, the web interface also displays and confirms the status of each channel. Each channel can be custom labeled to your specific function name. The controller operates on 12VDC or 12VAC at 500mA or our new AC121 global 12VDC switching power supply below. Factory assembled, tested, and ready to go! Even includes a Cat-5 cable!

VM201 8-Channel Remote Ethernet Controller, Factory Assembled & Tested

\$169.95

Air Blasting Ion Generator

Generates negative ions along with a hefty blast of fresh air, all without any noise! The steady state DC voltage generates 7.5kV DC negative at 400uA, and that's LOTS of ions! Includes 7 wind tubes for max air! Runs on 12-15VDC.



IG7 Ion Generator Kit

\$64.95

Broadband RF Preamp

Need to "perk-up" your counter or other equipment to read weak signals? This preamp has low noise and yet provides 25dB gain from 1MHz to well over 1GHz. Output can reach 100mW! Runs on 12 volts AC or DC or the included 110VAC PS. Assmb.



PR2 Broadband RF Preamp

\$69.95

Tri-Field Meter Kit

"See" electrical, magnetic, and RF fields as a graphical LED display on the front panel! Use it to detect these fields in your house, find RF sources, you name it. Featured on CBS's Ghost Whisperer to detect the presence of ghosts! Req's 4 AAA batteries.

THE GHOST DETECTOR!



TFM3C Tri-Field Meter Kit

\$74.95

Passive Aircraft Receiver

The hit of the decade! Our patented receiver hears the entire aircraft band without any tuning! Passive design has no LO, therefore can be used on board aircraft! Perfect for airshows, hears the active traffic as it happens! Available kit or factory assembled.



ABM1 Passive Aircraft Receiver Kit

\$89.95

Electret Condenser Mic

This extremely sensitive 3/8" mic has a built-in FET preamplifier! It's a great replacement mic, or a perfect answer to add a mic to your project. Powered by 3-15VDC, and we even include coupling cap and a current limiting resistor! Extremely popular!



MC1 Mini Electret Condenser Mic Kit

\$3.95

Laser Trip Sensor Alarm

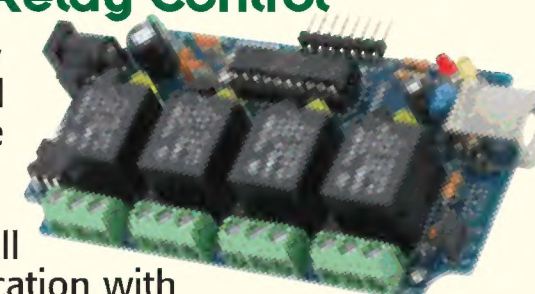
True laser protects over 500 yards! At last within the reach of the hobbyist, this neat kit uses a standard laser pointer (included) to provide both audible and visual alert of a broken path. 5A relay makes it simple to interface! Breakaway board to separate sections.

LTS1 Laser Trip Sensor Alarm Kit

\$29.95

4-Channel USB Relay Control

This professional quality USB relay controller and data acquisition module allows computer controlled switching of external devices, plus full bi-directional communication with the external world using the USB port of your computer. It is compatible with both Windows and Apple OS X, as well as various Linux flavors. When you plug it into your computer, it appears as a USB CDC device that creates a Virtual Serial (COM) port allowing easy communication with the board through any programming language that supports serial communications (VB, VB.NET, C#, C, C++, Perl, Java, etc).

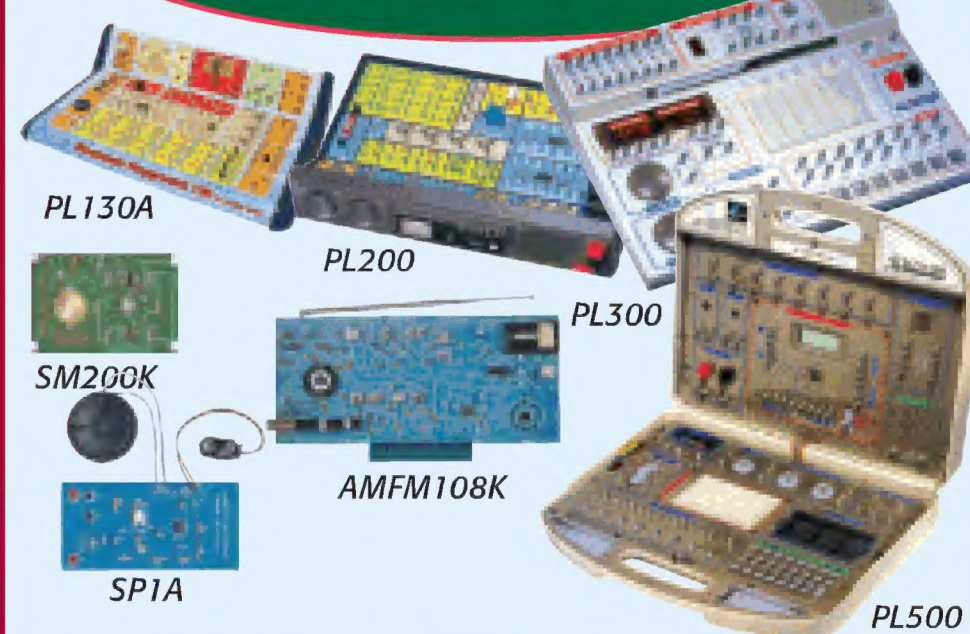


The controller features four onboard relay outputs with a current rating of 10A each. Also onboard is a 6-channel Input/Output interface, with each channel individually configurable as Digital Input, Digital Output, Analog Input (10-bit Resolution), or DS18B20 series Temperature Sensor. In Digital Input/Output modes, each channel can support a TTL compatible or ST input or a 5V output signal. In Analog Input mode, each channel can convert a voltage of between 0 to 5V into a 10-bit digital representation. Finally, in Temperature Sensor mode, each channel can be connected to a DS18B20 series Digital Temp Sensor.

UK1104 4-Ch USB Relay Interface Kit

\$59.95

The Learning Center!



Fun Electronic Learning Labs

- ✓ Learn and build!
- ✓ 130, 200, 300, & 500 in one labs!
- ✓ Practical through hole and SMT soldering labs!
- ✓ Integrated circuit AM/FM radio lab!
- ✓ Super comprehensive training manuals!

Starting out our "All in One" series, the PL130A, gives you 130 different electronic projects, together with a comprehensive 162 page learning manual. A great start for the kids...young and old! Next, check out the PL200, that gives you 200 very creative and fun projects, and includes a neat interactive front panel with 2 controls, speaker, LED display and a meter. From there, step up to our PL300, which gives you 300 separate electronic projects along with a 165 page learning and theory manual. The PL300 walks you through the learning phase of digital electronics. If you're looking for the ultimate lab kit, check out our PL500. It includes a whopping 500 separate projects, a 152 page starter course manual, a 78 page advanced course manual, and a 140 page programming course manual! The PL500 covers everything from the basics to digital programming!

If you are looking to either learn or hone up on your through hole or SMT soldering skills check our SP1A and SM200K Practical Soldering Labs. You will be a soldering master in no time!

We make it easy to learn IC's while at the same time, building a neat AM/FM radio with our AMFM108K AM/FM IC lab kit. You will have a blast AND learn!

PL130A	130-In-One Lab Kit	\$39.95
PL200	200-In-One Lab Kit	\$84.95
PL300	300-In-One Lab Kit	\$109.95
PL500	500-In-One Lab Kit	\$249.95
SP1A	Through Hole Soldering Lab	\$9.95
SM200K	SMT Practical Soldering Lab	\$22.95
AMFM108K	AM/FM IC Lab Kit & Course	\$36.95

GET THE NUTS AND VOLTS DISCOUNT!

Mention or enter the coupon code **NVRMZ12** and receive 10% off your order!

800-446-2295
www.ramseykits.com

Prices, availability, and specifications are subject to change. According to Robin we are not responsible for typos, stupid, printer's bleed, or tryptophan sleep comas from the turkey, and she certainly writes the rules! Visit www.ramseykits.com for the latest pricing, specials, terms and conditions. Copyright 2013 Ramsey Electronics®... so there!

RAMSEY ELECTRONICS®

590 Fishers Station Drive
Victor, NY 14564
(800) 446-2295
(585) 924-4560



CAD CONNECTED



Proteus Version 8
Launch Offer - SAVE 10%
Use NVL2013ATJ
When placing your order

PROTEUS DESIGN SUITE VERSION 8

Featuring a brand new application framework, common parts database, live netlist and 3D visualisation, a built in debugging environment and a WYSIWYG Bill of Materials module, Proteus 8 is our most integrated and easy to use design system ever. Other features include:

- Hardware Accelerated Performance.
- Unique Thru-View™ Board Transparency.
- Over 35k Schematic & PCB library parts.
- Integrated Shape Based Auto-router.
- Flexible Design Rule Management.
- Polygonal and Split Power Plane Support.
- Board Autoplacement & Gateswap Optimiser.
- Direct CAD/CAM, ODB++, IDF & PDF Output.
- Integrated 3D Viewer with 3DS and DXF export.
- Mixed Mode SPICE Simulation Engine.
- Co-Simulation of PIC, AVR, 8051 and ARM MCUs.
- Direct Technical Support at no additional cost.

labcenter  www.labcenter.com
Electronics

**Visit our website or
phone 866.499.8184
for more details**

Labcenter Electronics North America 411 Queen St, Newmarket, Ont. Canada L3Y 2G9
Email: info@labcenter-electronics.com Tel: 905.898.0665 Fax 905.898.0683